

SEMESTERARBEIT

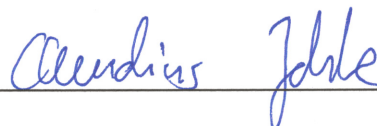
ENTWICKLUNG EINER PRÄDIKTIVEN VORSTEUERUNG FÜR EINEN NACHGIEBIG ANGEBRACHTEN ROBOTERENDEFFEKTOR

Freigabe:

Der Bearbeiter:

Unterschriften

Claudius Jehle



Betreuer:

Dr.-Ing. Friedrich Lange



Der Institutsdirektor

Prof. Dr. G. Hirzinger

Dieser Bericht enthält 101 Seiten, 47 Abbildungen und 15 Tabellen

Inhaltsverzeichnis

Inhaltsverzeichnis	I
Abbildungsverzeichnis	III
Tabellenverzeichnis	VI
Abkürzungsverzeichnis	VII
Verzeichnis der Formelzeichen	VIII
1 Einleitung	1
1.1 Ausgangssituation	2
1.2 Zielsetzung	4
2 Stand der Wissenschaft und Technik	5
2.1 Einführung	5
2.2 Methodenübersicht	6
2.3 Auswahl eines geeigneten Verfahrens	11
3 Theoretische Grundlagen	14
3.1 Grundlagen der Industrieroboter	14
3.1.1 Kinematik	14
3.1.2 Interne Regelstruktur	16
3.2 Prädiktive Steuerung	17
3.2.1 Modelansatz	17
3.2.2 Optimale Regelung	20
3.2.3 Berechnung der freien Systembewegung	23
3.2.4 Übertragungsfunktion und stationäre Genauigkeit	26
3.2.5 Stabilität	28
3.2.6 Einflüsse der Parameter	29
4 Realisierung	37
4.1 Systemmodellierung	37
4.1.1 Grundlagen	38
4.1.2 Roboterverhalten $R(z)$	39
4.1.3 Endeffektorverhalten $E(z)$	42
4.1.4 Gesamte Übertragungsfunktion	54
4.2 Regelungsansatz	55
4.2.1 Vorsteuerung	55

4.2.2	Integralregler	57
4.2.3	Gesamtstrecke	59
5	Experimente	60
5.1	Bewertungskriterien	60
5.2	Experimentenaufbau	61
5.3	Ergebnisse	63
6	Diskussion und Ausblick	73

Anhang A : Prädiktive Vorsteuerung

Anhang B : Mathematischer Anhang

Anhang C : Robotik

Anhang D : Systemidentifikation

Anhang E : Experimente

Anhang F : Sonstiges

Abbildungsverzeichnis

Abbildung 1: Montagesystem K2 mit KUKA KR180 Roboter (a), Endeffektor (b) und bewegter Fahrzeugkarosse (c).....	2
Abbildung 2: Robotersystem K1 am DLR.....	3
Abbildung 3: Abstrahierter Signalflussplan des Systems „Roboter & Werkzeug“ ohne (a) und mit (b) Vorsteuerung	4
Abbildung 4: Einfaches Signalflussschema einer adaptiven Steuerung	7
Abbildung 5: Verdeutlichung der prädiktiven Strategie – Erstellung eines Stellsignales u_+ , das den zukünftigen Fehler e_+ minimiert.....	10
Abbildung 6: Skizze des Roboters mit ausgelenktem Endeffektor mit wichtigen Koordinatensystemen und Punkten	14
Abbildung 7: Kette der Transformationen zwischen den verschiedenen Darstellungsformen. Alle Pfeile können auch umgekehrt werden.	16
Abbildung 8: Struktur der Roboterregelung und interne Lage der Endeffektorvorsteuerung	16
Abbildung 9: Signalflussplan des Regelgesetzes aus Gleichung (20).....	23
Abbildung 10: Signalflussplan mit Rückkopplung des Ausgangs.....	24
Abbildung 11: Signalflussplan mit Einspeisung eines prädizierten Systemausgangs.....	25
Abbildung 12: Signalflussplan mit Einspeisung eines angereicherten prädizierten Systemausgangs.....	26
Abbildung 13: Stellgrößen für einen Sprung in einem schwach-gedämpften System für drei verschiedene Parametersätze	31
Abbildung 14: Antworten für einen Sprung in einem schwach-gedämpften System für drei verschiedene Parametersätze.....	32
Abbildung 15: Stellgrößen für einen Sprung in einem schwach-gedämpften System für verschiedene Prädiktionshorizonte ($N_c=1$, $q^k=1,00$, $r^k=0,00$)	34
Abbildung 16: Systemausgang für einen Sprung in einem schwach-gedämpften System für verschiedene Prädiktionshorizonte	34
Abbildung 17: Stellgrößen für einen Sprung bei Variation von $N_p=N_c$	35
Abbildung 18: Systemausgang für einen Sprung bei Variation von $N_p=N_c$	35
Abbildung 19: Fotografie des Endeffektors K2	37

Abbildung 20: Umdefinition des Systemausgangs zum SCP.....	38
Abbildung 21: Skizze zur Verdeutlichung der Lagen von TCP und SCP im (a) unausgelenkten Nominalfall und (b) ausgelenkten Fall	39
Abbildung 22: Detailansicht des dynamischen Systems „Roboter“	40
Abbildung 23: Gemessene Übertragungsfunktion von A6 des Roboters in Abhängigkeit von \$FILTER für $f \geq 3,00$ Hz	40
Abbildung 24: Darstellung der Auslenkung des Endeffektors bei nicht-waagerechter Lage des SCP0	45
Abbildung 25: Anpassungsergebnis der Jacobi-Matrix für den Winkel α mit dem Faktor $J^{\alpha} = -20,2 \cdot 10^{-3}$	46
Abbildung 26: Lineares Chripsignal mit $f_0 = 0,00$ Hz und $k = 0,20$ Hz \cdot s $^{-1}$	50
Abbildung 27: Gemessene FFT-Daten zur Identifikation der Übertragungsfunktion $z \rightarrow \beta$ (K1 ; $K_{z\beta} = 4,73 \cdot 10^{-3}$; $F_{\beta} = 6,59$ Hz; $D_{\beta} = 29,5 \cdot 10^{-3}$)	51
Abbildung 28: Gemessene und identifizierte Systemantwort für ein Chripsignal $z \rightarrow \beta$ im Zeitbereich (vgl. Abbildung 27) (K1)	53
Abbildung 29: Gemessene und identifizierte Systemantwort für die Bahn impulse10.bahn $z \rightarrow \beta$ (K1 , \$FILTER=0)	53
Abbildung 30: Signalflußplan des gesamten Systemmodells H(z)	55
Abbildung 31: Signalflussplan des geschlossenen Regelkreises mit Vorbehandlungsbaustein (Φ für stat. Genauigkeit \rightarrow Kap. 3.2.4)	56
Abbildung 32: Gesamter Signalflussplan	59
Abbildung 33: Verdeutlichung zweier Kriterien zur Bewertung der Steuergüte (Fehlerfläche FF und max. Bahnfehler BF_{\max})	60
Abbildung 34: Freiheitsgrad β der Bahn impulse10 mitsamt Geschwindigkeits- und Beschleunigungsverlauf (linear bzw. konstant)	62
Abbildung 35: Auslenkung von α (rot) aufgrund einer Anregung des gekoppelten Freiheitsgrades y (blau, nicht maßstäblich) für K1	62
Abbildung 36: Fehlervergleich exemplarisch für α (impulse-Bahn) zwischen Simulation und Realität	64
Abbildung 37: Vergleich der Sollstellgröße und der tatsächlich ausgeführten Stellgröße durch Einfluss der Kompensationsvorsteuerung ($\alpha \rightarrow y$)	65
Abbildung 38: Vergleich der Stellgrößen (SCP0-Lage β) zwischen Steuergesetz mit und ohne Berücksichtigung des Mittelwertfilters	65

Abbildung 39: Tatsächlich ausgeführte Stellgrößen (Lage des SCP0 für β ; Amplitudengrößenordnung beachten!).....	66
Abbildung 40: TCP-Schwingungen für β mit und ohne prädiktive Steuerung	68
Abbildung 41: Stellgröße und Ausgang für β	69
Abbildung 42: Vgl. der Systemausgänge ($y \rightarrow \alpha$) für keine Vorsteuerung (schwarz) und Parametersatz 25 (rot gemessen, blau simuliert).....	69
Abbildung 43: Ausgleich des statischen Fehlers 1. Ordnung (für β).....	70
Abbildung 44: Veranschaulichung der Dimensionen und Besetzung der grundlegenden Systemmatrizen.....	3
Abbildung 45: Darstellung der Drehreihfolge mit α um z, β um y und γ um x	1
Abbildung 46: Schematische Darstellung des ausgelenkten Endeffektors	3
Abbildung 47: Frequenzgang der Übertragungsfunktionen $z \rightarrow z$ (li.) und $y \rightarrow \alpha$ (re.) für Roboter K2	3

Tabellenverzeichnis

Tabelle 1: Bewertungsschema der verschiedenen Steuerungsalgorithmen	12
Tabelle 2: Bezeichnungen der Systemkomponenten	17
Tabelle 3: Einflüsse der Strafmatrizen auf die Stellgrößen	30
Tabelle 4: Besetzung der Strafmatrizen für drei exemplarische Fälle.....	32
Tabelle 5: Roboterkonfigurationen.....	61
Tabelle 6: Veränderung des (a) maximalen Bahnfehlers \underline{BE}_{\max} und (b) der max. Fehlerfläche \underline{FE} für 4 Parametersätze.....	67
Tabelle 7: Parametersätze der Experimente aus Tabelle 6.....	68
Tabelle 8: Sonstige Matrixdimensionen	3
Tabelle 9: Schwingungsparameter von \mathbf{E}_{dyn} für K1	1
Tabelle 10: Übertragungsparameter K_{mn} von \mathbf{E}_{dyn} für K1 ; $m=10^{-3}$, eingeklammerte Werte unsicher, $\approx 0,00$	1
Tabelle 11: Schwingungsparameter von \mathbf{E}_{dyn} für K2	2
Tabelle 12: Übertragungsparameter K_{mn} von \mathbf{E}_{dyn} für K2 ; $m=10^{-3}$, eingeklammerte Werte unsicher, ≈ 0	2
Tabelle 13: Überblick über alle Parametersätze (Feld leer→s. Feld darüber)	1
Tabelle 14: Versuchsprotokoll	4
Tabelle 15: Format der pose-Dateien	5

Abkürzungsverzeichnis

Technische Abkürzungen

ARX	AutoRegressive model with eXogenous input	
DLR	Deutsches Zentrum für Luft- und Raumfahrt	
EE	Endeffektor	
FFT	Fast FOURIER Transformation	
FIR	finite impulse response (Filter) ($\rightarrow IIR$)	
GPC	General Predictive Control(ler)	
IIR, I ² R	infinite impulse response (Filter) ($\rightarrow FIR$)	
iwb	Institut für Werkzeugmaschinen und Betriebswissenschaften	
KMS	Kraft-Momenten-Sensor	
LMS	Least Mean Square (Algorithmus)	
MIMO	Multiple-Input Multiple-Output (System)	
oBdA	ohne Beschränkung der Allgemeinheit	
RM	Robotik- und Mechatronik-Institut des \rightarrow DLR	
SCP(0)	Spring Center Point (nominal)	(<i>Federmittelpunkt</i>)
SISO	Single-Input Single-Output (System)	
TCP(0)	Tool Center Point (nominal)	(<i>Werkzeugmittelpunkt</i>)
ÜF	Übertragungsfunktion	
ZPETC	Zero-Phase Error Tracking Control	

Sonstige Abkürzungen

bspw.	beispielsweise
ggf.	gegebenenfalls
i. A.	im Allgemeinen
resp.	respektive (\rightarrow bspw.)
z.B.	zum Beispiel
z.T.	zum Teil

Verzeichnis der Formelzeichen

Allgemeine Größen

n, n	allgemeiner Skalar, allgemeine Größe
\underline{v}	allgemeiner Vektor
\underline{W}	allgemeiner Großvektor, Vektor von Vektoren
\mathbf{M}, \mathcal{M}	allgemeine Matrix
$(\cdot)_{+/-}$	allgemeine Größe, enthält Informationen über die Zukunft/Vergangenheit
$f(\cdot)$	allgemeine Funktion von ...
$\mathbf{0}_{p \times q}$	$(p \times q)$ -Matrix mit allen Elementen 0
$\mathbf{1}_{p \times q}$	$(p \times q)$ -Matrix mit den Elementen der Hauptdiagonalen 1, ansonsten 0
$n\check{\mathbf{I}}_p$	p -fache Übereinanderstellung von $\mathbf{1}_{n \times n}$ Matrizen (n fehlt $\Leftrightarrow n=M$)
\mathcal{R}	reelle Zahlen
\mathbb{Z}	ganze Zahlen

Kinematische und geometrische Größen

\underline{q}_A	6-dimensionaler Vektor der Achswerte für die Lage von Punkt A
${}^K\underline{p}_{A,B}$	6-dimensionaler Lagevektor von A nach B im Koordinatensystem K
${}^A\mathbf{T}_B$	homogene Transformationsmatrix von Objekt B relativ zu Objekt A
M	Systemfreiheitsgrad (allgemein 6)
\mathbf{J}_0	JACOBI-Matrix des Endeffektors in Nulllage

Systemtheoretische und regelungstechnische Größen

τ	Abtasttaktzeit (hier $\tau=12ms$)
z	1-dimensionaler diskreter Zeitverschiebeoperator
ζ	M-dimensionaler diskreter $Z\sim$ (\rightarrow Anhang B)
$\mathcal{F}\{\cdot\}$	FOURIER-Transformationsoperator
h, D	Totzeit
\underline{u}	Stellsignal

$\underline{y} / \check{y} / \dot{y}$	Systemausgang, Ist-Zustand (gemessen/modellbasiert/gemischt)
\underline{w}	Sollgröße, Solltrajektorie
\underline{e}	Fehler, Differenz zwischen Sollwert und Ausgang
$\underline{u}, \underline{p}$	vektorielle Größe (Lage) im diskreten LAPLACE-Bereich
$\Sigma(z)$	Übertragungsfunktion der realen Strecke
$H(z)$	Übertragungsfunktion des Streckenmodells
$\mathcal{V}_{A\dots Z}$	Gütefunktional bzgl. den Strafmatrizen $\mathbf{A}\dots\mathbf{Z}$

Größen der prädiktiven Vorsteuerung

N_p	Prädiktionshorizont
N_c / N_w	Steuerungshorizont
\mathbf{Q}	Fehlerstrafmatrix
\mathbf{R}	Stellgrößenstrafmatrix
ε	<i>Skalierungsfaktor der Matrix \mathbf{R}</i>
$\mathbf{\Gamma}$	Übertragungsfunktion/allg. Bezeichnung der prädiktiven Vorsteuerung
$\mathbf{C}_{+/-}$	Systemdynamikmatrix der zukünftigen/vergangenen Stellgrößen
\mathcal{D}	Systemdynamikmatrix der vergangenen Ausgänge
\mathbf{G}_+	\mathbf{Q} -gewichtete Systemdynamikmatrix \mathbf{C}_+
\mathbf{K}	Übertragungsmatrix der prädiktiven Vorsteuerung
$\mathcal{K}_{W/U/Y}$	zusammengefasste Vorsteuerungsmatrizen für Eingang/Stellgrößen/Ausgang
Φ	Korrekturmatrix für stationäre Genauigkeit

Sonstige Größen

j	komplexe Einheit ($j^2=-1$)
$\underline{BF}[k]$	Bahnfehler
\underline{BF}_{max}	maximaler Bahnfehler
\underline{FF}	normierte Fehlerfläche
$\$VAR$	Variable der KUKA-Steuerung

1 Einleitung

In den Montagehallen von morgen werden immer mehr Prozesse hoch-automatisiert durch Industrieroboter ausgeführt. Roboter bieten Vorteile in der Präzision, Geschwindigkeit, Arbeitszeit und der Wiederholungsgenauigkeit der durchgeführten Tätigkeiten. Jedoch sind manche dieser Vorteile durch externe Faktoren eingeschränkt. Beispielsweise müssen Montageroboter an Montagestraßen warten, bis das von ihnen bearbeitete Bauteil (z. B. eine Fahrzeugkarosse) an- und zur Ruhe gekommen ist, bevor sie ihre Tätigkeit automatisiert beginnen können. Menschen können mithilfe der Wahrnehmung ihrer Umwelt Ungenauigkeiten und Ungereimtheiten erkennen, analysieren und sich auf die Behebung des Problems einstellen. Daher werden immer mehr Roboter montagesysteme mit Sensoren zur Umweltwahrnehmung ausgestattet.

Das betrachtete Projekt wurde in einer Zusammenarbeit zwischen dem *Institut für Werkzeugmaschinen und Betriebswissenschaften (iwb)* der Technischen Universität München und dem *Institut für Robotik und Mechatronik (RM)* des *Deutschen Zentrums für Luft- und Raumfahrt (DLR)*, sowie weiteren Industriepartnern (*Schunk GmbH & Co. KG* (Lauffen), *Quiss GmbH* (Puchheim), *Sturm Maschinenbau GmbH* (Salching), *KUKA Roboter GmbH* (Augsburg) und der *BMW AG* (München)) bearbeitet. Nach Ende der Kooperation im Jahr 2007 wird das Projekt vom RM-Institut und der *KUKA Roboter GmbH* fortgeführt. Es wurde ein Werkzeug zur automatisierten Montage von Rädern an geförderten Fahrzeugkarossen entwickelt. Es wird von einem KUKA Industrieroboter bewegt und greift eigenständig ein neues Rad, fährt in Warteposition und erkennt mithilfe einer Kamera die Radnaben der ankommenden Karossen. Das Rad wird automatisch an die Nabe angeschraubt. Dabei kann sich die Karosse noch in Bewegung befinden, und die Nabe muss nicht in einer besonderen Weise ausgerichtet sein (es kann also z. B. ein Lenkwinkel eingeschlagen sein).

Allgemein kann das System aber auch für andere Anwendungen verwendet werden, in denen eine Montage an einem bewegten Objekt vollführt werden muss. Denkbar sind Anwendungen in der industriellen Fertigung, jedoch auch in der Medizintechnik (Operation am schlagenden Herzen) und in der Raumfahrt.

Es besteht allerdings noch weitergehender Forschungsbedarf. Insbesondere sind Schwingungen des Werkzeuges aufgrund der Roboterbewegung ein ungelöstes Problem. Dieser Aufgabe wird in dieser Arbeit versucht beizukommen.

1.1 Ausgangssituation

Ausgangspunkt der Arbeit ist das Montagesystem, wie es in der Einleitung angesprochen wurde. Dabei handelt es sich um einen KUKA KR180 Industrieroboter, an den über (flexible) Kraft-Momenten-Sensoren das Schraubwerkzeug (im Folgenden auch allgemein als *Endeffektor* bezeichnet) angeflanscht ist. Die Kamera, deren Blickrichtung durch das Radloch geht, und die zugehörige Regelung gewährleisten die korrekte Ausrichtung und Ansteuerung des Roboters, sodass Radschrauben und Nabenbohrungen fluchten. Ein Bild des Gesamtsystems ist in Abbildung 1 gegeben. Dieses System wird im Folgenden mit **K2** bezeichnet.

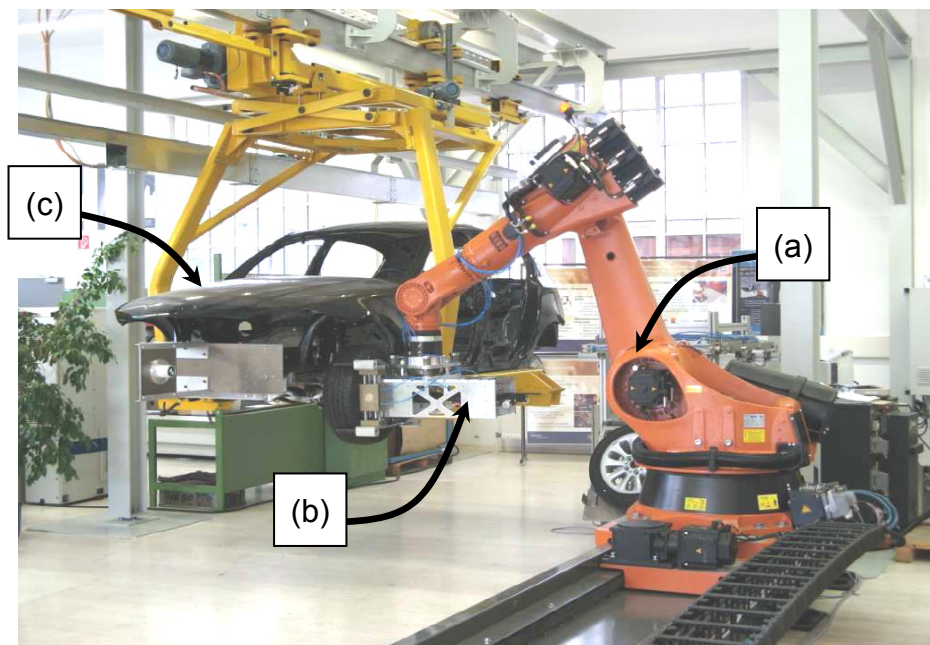


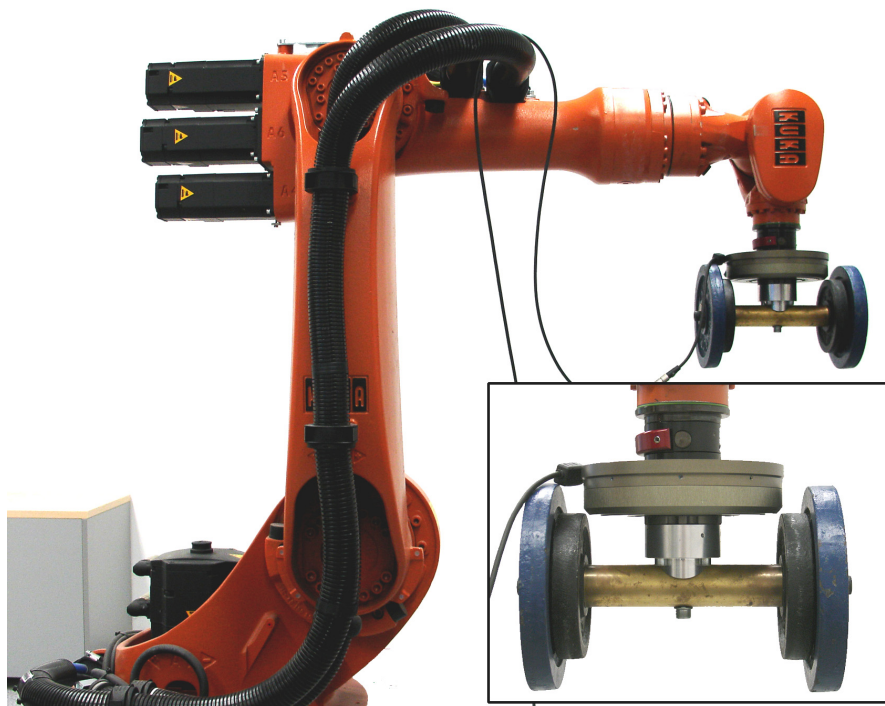
Abbildung 1: Montagesystem **K2** mit KUKA KR180 Roboter (a), Endeffektor (b) und bewegter Fahrzeugkarosse (c)

In vorangegangenen Arbeiten wurde die Endeffektormechanik und -funktionalität (insbesondere die Pressluftschraubendreher und die Kraft-Momenten-Sensoren) entwickelt und gefertigt, die sensor- bzw. kamera-gesteuerte Annäherung des Werkzeugs implementiert und der kraft-geregelte Anschraubvorgang der Räder an die Radnaben entwickelt.

Es wurde sich für nachgiebige (gegenüber steifen) Kraft-Momenten-Sensoren entschieden, weil durch sie die Anforderungen an die Regelung des *Gesamtsystems* geringer ausfallen. Es wurde so erreicht, dass der Roboter weniger sensibel gegenüber Stößen und Störungen (z.B. ein Nachschwingen der Fahrzeugkarosse) ist

und langsamer reagieren muss. Ein großer Nachteil der Nachgiebigkeit jedoch ist die Anregung von Schwingungen des Werkzeuges. Durch das Bewegen des Endeffektors wird dieser zum Schwingen angeregt, was Konsequenzen für die Bahntreue hat. Durch die Schwingungen wird die Positioniergenauigkeit des Werkzeuges (insb. dessen Arbeitspunktes, des *tool center points*, kurz TCP) stark beeinträchtigt. In dieser Arbeit wird versucht, durch geeignete Ansteuerung des Roboters diesen Schwingungen entgegenzuwirken. Allgemeiner gesagt wird versucht, trotz vorhandener Flexibilität zwischen Roboter und Endeffektor ein möglichst präzises Abfahren einer vorgegeben Bahn des Endeffektors zu gewährleisten.

Da der Steuerungsalgorithmus zuerst an einem leichter handhabbaren System¹ getestet werden sollte, wurde am DLR ein kleinerer KUKA KR16 Roboter mit einem Pseudo-Endeffektor versehen, an dem praktisch alle Tests vorgenommen wurden. Diese Konfiguration wird mit **K1** bezeichnet und ist in Abbildung 2 dargestellt.



*Abbildung 2: Robotersystem K1 am DLR
(unten rechts Detailansicht des Pseudo-Endeffektors)*

¹ Der Endeffektor von **K2** ist durch zwei Kraft-Momenten-Sensoren mit integrierter Feder-Dämpfer-Kombination angeflanscht. Dieses System weist u.a. Hystereseeffekte auf. In **K1** hingegen wurde ein weitgehend linearer KMS verbaut.

1.2 Zielsetzung

Durch die direkte Einspeisung der Solltrajektorie des Werkzeugs in den Roboter werden Schwingungen induziert, die ein bahntreues Positionieren des Werkzeugs verhindern. Ziel war es deshalb, eine Softwarevorsteuerung zu entwickeln, die den Roboter so ansteuert, dass trotz der vorhandenen Flexibilität ein möglichst bahntreues Abfahren ermöglicht wird. Signalflußtechnisch ist dies in Abbildung 3 gezeigt.

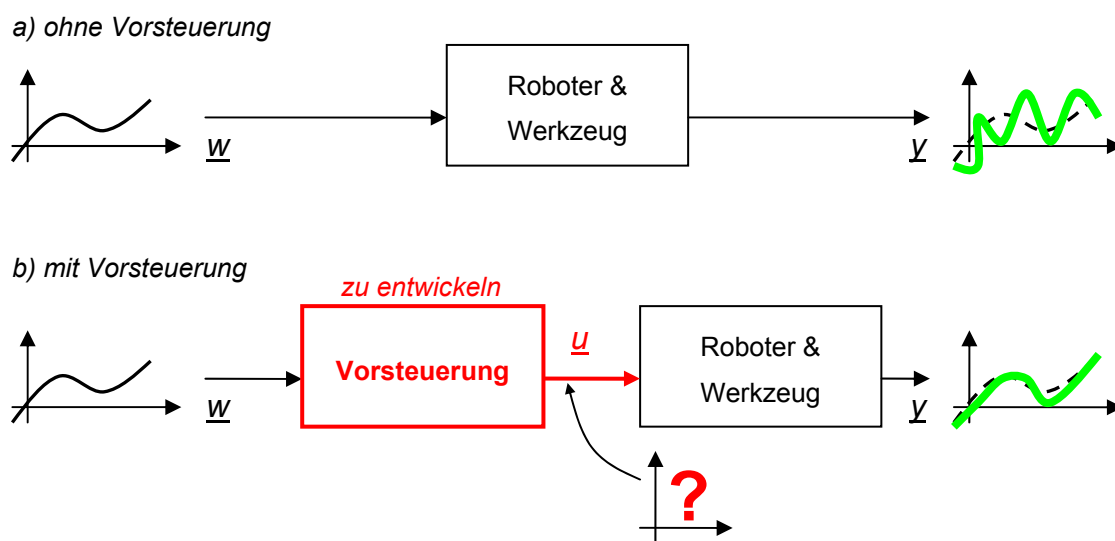


Abbildung 3: Abstrahierter Signalflussplan des Systems „Roboter & Werkzeug“ ohne (a) und mit (b) Vorsteuerung

Für die Realisierung dieser Vorsteuerung mussten einige Bedingungen berücksichtigt werden. Diese sind unter anderem:

1. Einbindung der Vorsteuerung in ein vorhandenes, hierarchisches System (siehe dazu auch Abbildung 8)
2. insbesondere Bereitstellung von aktuellem *und* zukünftigen Stellgrößen für die nachgeordnete prädiktive Steuerung des Roboterarms
3. Einhaltung der Rechenzeit (Abtastzeit $\tau=12ms$)
4. Weitgehend unbekannte Systemdynamik des Endeffektors
5. Berechnung im kartesischen Raum (nicht auf Achsbasis)
6. Berücksichtigung im Betrieb korrigierter Soll-Trajektorien

2 Stand der Wissenschaft und Technik

2.1 Einführung

Zur Gewährleistung von Regelzielen gibt es in der Allgemeinen Form zwei grundlegende Strategien. Die eine ist die Rückkopplung des Ausgangssignals und der Vergleich mit einem Sollwert. Weichen sie voneinander ab, so wird eingegriffen. In diesem Falle spricht man von einer (rückkoppelnden) *Regelung*, im englischen auch von *feedback control*. Ein Beispiel ist die Anpassung der Fahrzeuggeschwindigkeit an eine Sollgeschwindigkeit (z.B. innerorts 50 kmh^{-1}) durch den Fahrer, der über das Tachometer (Messeinrichtung/Rückkopplung) eine Rückmeldung über die aktuelle Istgeschwindigkeit erhält (Fahrer ist der Regler). Ein Vorteil ist, dass das zu regelnde System nicht genau bekannt sein muss. Jedoch ergibt sich nachteilig eine mögliche Instabilität durch die Rückkopplung, eine zwangsweise Messbarkeit des Istzustandes und ein tendenziell langsames Verhalten. Wird ein Sollwert eingestellt, ohne das Ergebnis mit dem tatsächlichen Istzustand zu vergleichen, so spricht man hingegen von einer *Steuerung* bzw. im englischen von *feedforward control*. Dies scheint auf den ersten Blick natürlich nachteilig zu sein, jedoch ist eine Steuerung oft ausreichend. Beispielhaft kann ein Elektromotor genannt werden, dessen Drehzahl durch einen bestimmten Vorwiderstand eingestellt wird. Es wird darauf vertraut, dass die Auslegung des Vorwiderstandes korrekt ist, und der Motor die gewünschte Drehzahl liefert. Eine Steuerung ist schnell, allgemein stabil und einfach verständlich. Jedoch ist zwingend Wissen über das zu steuernde System nötig¹. Grundlegende Literatur zu diesem Thema ist bspw. [1], [2] oder [3].

Aufgrund der Nachteile einer Regelung, insbesondere der langsameren Reaktionszeit und der möglichen Instabilität bei komplexeren Systemen, soll hier ein Steuerungsansatz verfolgt werden. Im Folgenden werden unterschiedliche Konzepte einer Vorsteuerung vorgestellt, deren Vor- und Nachteile beleuchtet und abschließend eine geeignete Methode Faktoren ausgewählt.

¹ Hiermit sollte nur der grundlegende Unterschied zwischen reiner Vorsteuerung und einer rückkoppelnden Regelung verdeutlicht werden. Im Folgenden wird öfter unkorrekter Weise von einer „Regelung“ die Rede sein; gemeint ist jedoch, falls keine andere Bemerkung gemacht wird, stets eine Steuerung. Im Englische existiert mit „control“ gar keine begriffliche Unterscheidung.

2.2 Methodenübersicht

Inverse Strecke

Der einfachste Ansatz für eine Steuerung ist das Vorschalten einer *inversen Strecke*. Sei Σ ein dynamisches System, y der Systemausgang, u der Systemeingang und w die Sollgröße (je im LAPLACE-Bereich), so liefert die Vorschaltung von Σ^{-1} offensichtlich eine perfekte Sollgrößenfolge mit $y=w$:

$$y = \Sigma \cdot u \xrightarrow{u = \Sigma^{-1} \cdot w} y = \Sigma \cdot (\Sigma^{-1} \cdot w) = w \quad (1)$$

Lässt sich das System invertieren, so kann die inverse Strecke als einfache und optimale Lösung des Problems angesehen werden. Dazu muss aber offensichtlich gewährleistet sein, dass die Strecke überhaupt mathematisch invertierbar ist (sie muss bspw. quadratisch sein). Ist dies gegeben, muss zudem gewährleistet sein, dass die Inverse *stabil* ist. Ist Σ *nicht-minimalphasig*, d. h., existieren instabile Übertragungsnullstellen (im kontinuierlichen Fall Nullstellen mit einem positiven Realteil bzw. im diskreten Fall Nullstellen mit einem Betrag ≥ 1), so ist die Inverse instabil. Bei MIMO-Systemen („multiple input–multiple output“, zu Deutsch „System mit mehreren Ein- und Ausgängen“, Mehrgrößensystem) können durch interne Kopplungen instabile Nullstellen auftreten, obwohl die einzelnen Übertragungsfunktionen allesamt minimalphasig sind.

Durch die differenzierenden Eigenschaften des hier behandelten Systems ist eine einfache Invertierung daher als kritisch anzusehen (d.h., ein Teil der Übertragungsnullstellen wird auf dem Einheitskreis liegen; Details siehe auch Kapitel 4.1.3.2). Es wird deshalb im Folgenden als nicht-minimalphasig angesehen.

Approximierung der inversen Strecke

Eine weit verbreitete Möglichkeit, die direkte Invertierung zu vermeiden, ist die *Annäherung* der Inversen. Sei diese mit Z beschrieben, so ergibt sich aus (1):

$$y = (\Sigma \cdot Z) \cdot w \rightarrow w \Leftrightarrow Z \rightarrow \Sigma^{-1} \quad (2)$$

Beispielsweise kann Z so angepasst werden, dass es das Verhalten von Σ^{-1} nur in bestimmten (Frequenz-)Bereichen nachbildet. Dabei kann darauf geachtet werden, dass Z stabil bleibt. FIR-Filter (*finite impulse response*) basieren auf der Impulsantwort eines Systems und sind inhärent stabil [4]. Sie können nahezu

beliebiges Systemverhalten abbilden und eignen sich deshalb sehr gut für die Approximierung einer inversen Strecke. Jedoch sind für ein solches Filter umso mehr Koeffizienten erforderlich, je länger die Impulsantwort ist. Bei schwach-gedämpften Systemen mit vergleichsweise hoher Abtastrate ergeben sich sehr viele Koeffizienten, wie aus folgendem Beispiel entnommen werden kann.

Beispiel 1: Impulsantwort eines schwach-gedämpften Systems

Man lässt ein ausgelenktes Feder-Dämpfer-Masse-System zum Zeitpunkt 0 frei und misst die Auslenkung über die Zeit. Das System kommt zu einer Zeit $t_{end}=17,00s$ zur Ruhe. Die Messapparatur habe eine Abtastfrequenz von $f_a=100Hz$, woraus sich eine Abtastzeitkonstante $\tau=10ms$ ergibt. Um das Schwingungsverhalten dieses Systems durch einen FIR-Filter nachzustellen, sind

$$N = \frac{17,00s}{10 \cdot 10^{-3}s} = 1700$$

Koeffizienten nötig. Zum Vergleich: Stellt man dasselbe Verhalten durch ein IIR-Filter (*infinite impulse response*) mit einer internen Rückkopplung dar, so sind nur ca. 4-6 Koeffizienten nötig.

Es ist leicht einzusehen, dass die Anzahl der Koeffizienten und damit der Aufwand an Speicher und Rechenzeit *nochmals* drastisch zunimmt, sobald es sich um ein verkoppeltes Mehrgrößensystem handelt. Nichtsdestotrotz findet eine Approximation der Inversen durchaus Anwendung. Als sehr geeignet hat sich die Approximation durch *adaptive Anpassung*, beispielhaft nach folgendem Schema, erwiesen:

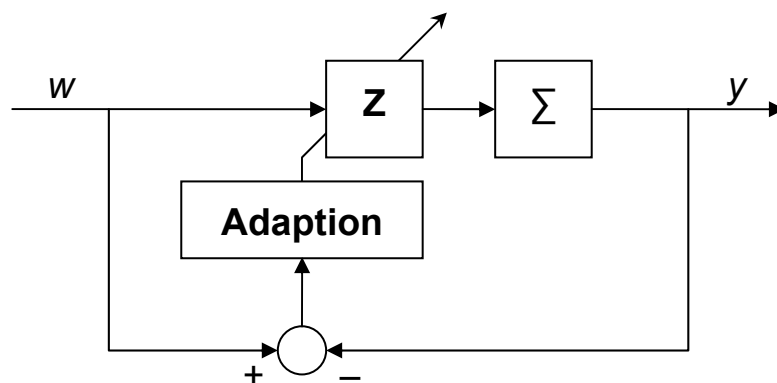


Abbildung 4: Einfaches Signalflussschema einer adaptiven Steuerung

Der Adaptionsmechanismus stellt die Koeffizienten von \mathbf{Z} zur *Laufzeit* so ein, dass der Fehler zwischen Sollgröße w und Istgröße y minimal wird, sie sich somit bestmöglich gleichen. Würde für alle Frequenzen, die in w enthalten sind, $w-y=0$ gelten, so wäre \mathbf{Z} tatsächlich die Inverse von Σ . Die Anzahl der Koeffizienten von \mathbf{Z} muss vorgegeben werden. Einen guten und kurzen Überblick, auch zu nicht-minimalphasigen Systemen, bietet WIDROW et al. in [5]. Anzumerken ist, dass ein Adaptionsmechanismus den Regelkreis durch die Rückkopplung schließt, und man deshalb genaugenommen von einer Regelung sprechen muss.

Nicht-kausale Invertierung

Als eine weitere Möglichkeit kann eine nicht-kausale Inversion erfolgen, wie sie u.a. von SILVERMAN [6] vorgeschlagen wurde. Der grundlegende Gedanke ist, dem Problem der resultierenden Instabilität der Inversen durch Kenntnis der vergangenen und zukünftigen Trajektorie entgegenzukommen. Dieses mathematisch anspruchsvolle Konzept erstellt ein Eingangssignal derart, dass die internen (instabilen) Zustände der Inversen entlang einer instabilen Mannigfaltigkeit auf die Nulldynamik-Mannigfaltigkeit getrieben werden [7]. Dazu ist die genaue Kenntnis der Solltrajektorie von Anfang *bis Ende* (\rightarrow *nicht-kausal*) nötig, was eine Anwendung in einem System mit Trajektorienplanung bzw. –korrektur, wie sie hier vorhanden ist, einschränkt (s. dazu auch [8/§2.3]). Diesem Problem wurde in [8] teilweise begegnet. Jedoch muss dazu einerseits die Invertierung online vorgenommen werden, was je nach Größe des Systems zu einem hohen Rechenaufwand führen kann. Andererseits muss die Vorschauzeit in Abhängigkeit von der Nullstellenlage gewählt werden, um ein ausreichend gutes Steuerergebnis zu erhalten. Durch den differenzierenden Charakter des Systems ist dadurch eine hohe Vorschauzeit nötig. Da es sich jedoch immer noch um eine modellbasierte Steuerung handelt, hängt die Steuergüte grundsätzlich von der Qualität des Modells ab. Es ist fraglich, ob sich der vergleichsweise hohe Aufwand der Implementierung für ein 6-Freiheitsgrade-System lohnt. Mit dem Problem der Modellunsicherheiten beschäftigt sich DEVASIA in [9].

WEN und POTSAID führen in [10] zusätzlich an, dass nicht-kausale Invertierung abhängig von den Pol-Nullstellen-Lagen zu numerischen Problemen führen kann.

Zero-Phase Error Tracking Control (ZPETC)

Der Grundgedanke dieser Steuerungsstrategie ist einfach. Man finde ein Vorfilter, das einen verschwindenden Phasenversatz (*zero-phase*) des geschlossenen Regelsystems erwirkt. Daraus resultiert eine *zeitlich* perfekte Sollgrößenfolge – der

gravierende Nachteil ist jedoch, dass der statische Übertragungsfaktor für nicht-minimalphasige Systeme zwangsweise ungleich 1 ist. Für ein SISO-System lässt sich ZPETC einfach herleiten (s. [10/§II.B]). Seien $Z^{(u)}$ bzw. $Z_z^{(s)}$ die (in)stabilen Zählerpolynome und N das Nennerpolynom der Übertragungsfunktion $G(z^{-1})$ nach folgender Form:

$$G(z^{-1}) = \frac{Z^{(s)}(z^{-1}) \cdot Z^{(u)}(z^{-1})}{N(z^{-1})} \quad (3)$$

Dann ergibt sich das Vorfilter $F(z)$, welches Null-Phasenverschiebung gewährleistet:

$$F(z) = \left(\frac{N(z^{-1})}{Z^{(s)}(z^{-1}) \cdot Z^{(u)}(1)} \right) \cdot \left(\frac{Z^{(u)}(z^{-1})}{Z^{(u)}(1)} \right) \quad (4)$$

Dem Problem des stationären Übertragungsfaktors ungleich 1 wurde zwar in verschiedenen Veröffentlichungen begegnet (bspw. [11]), jedoch teilweise mit erheblichem Aufwand und nur unter Kombination verschiedener Techniken (u.a. auch prädiktive Verfahren).

Es existiert ein Konzept von RUZBEHANI [12] für SISO-Systeme, dass einen Verstärkungsfaktor von 1 ermöglicht, aber eine Totzeit in Abhängigkeit der Nullstellenlage enthält. Da das hier behandelte System Nullstellen auf dem Einheitskreis hat (\leftrightarrow differenzierendes System), würde sich eine sehr große Totzeit ergeben.

Inputshaping

Ein weiterer Weg ist die Modifikation des Eingangssignals derart, dass mögliche Schwingungen gar nicht erst angeregt werden. Man spricht von Eingangssignalgenerierung oder engl. Inputshaping. Dieser Ansatz wurde in einer vorangegangenen Arbeit schon verfolgt hat sich als nicht praktikabel erwiesen. Details s. [13].

Prädiktive Steuerung

Das Konzept der prädiktiven Steuerung basiert auf der Erstellung eines Stellsignals, welches versucht, den zukünftigen Fehler zwischen Soll- und Istzustand des Systems zu minimieren. Der zukünftige Fehler (in Abbildung 5 mit e_+ bezeichnet) wird durch Vergleich der (bekannten) zukünftigen Trajektorie (w_+) mit dem

vorhergesagten (\leftrightarrow *prädiziert*) Systemausgang y_+ errechnet. Nun lässt sich nach dem Prinzip der optimalen Regelung ein Gütefunktional mit dem Fehler erstellen, aus dem ein Stellsignal u_+ abgeleitet werden kann, welches diesen Fehler innerhalb eines gewissen Horizontes minimiert.

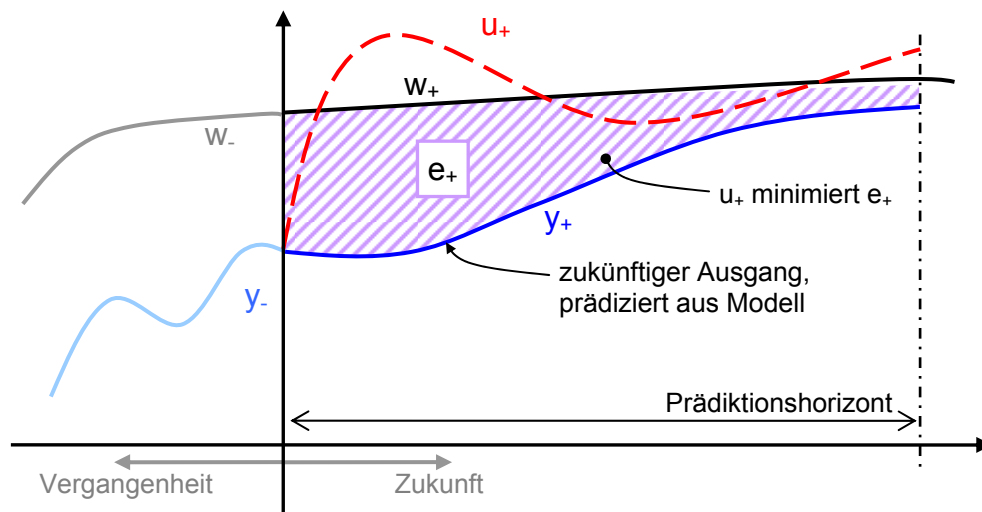


Abbildung 5: Verdeutlichung der prädiktiven Strategie – Erstellung eines Stellsignales u_+ , das den zukünftigen Fehler e_+ minimiert

Prädiktive Steuerungsalgorithmen bieten verschiedene Vorteile:

- weite Verbreitung in Industrie und Forschung, etablierte Technik
- Einfache Implementierung auch für Mehrgrößensysteme
- Nicht-minimalphasige Systeme durch geeignete Parameterwahl beherrschbar
- Durch das Optimierungsproblem lassen sich unterschiedliche Aspekte der Steuerung flexibel gewichten (bspw. „Minimierung des Fehlers unter allen Umständen“ gegenüber „Minimierung bei gleichzeitiger Gewichtung der Stellgrößen“)
- Durch den prädiktiven Charakter kann die nach Punkt 2 aus Kapitel 1.2 wichtige Folge *zukünftiger* Stellgrößen gewonnen werden
- Erweiterbarkeit um andere Techniken (Adaption, Rückkopplung, harte Beschränkungen)
- Prädiktion weitgehend unabhängig vom Systemmodell ([14/§3])

Aus einer Impulsantwort-Darstellung entstand ursprünglich das *model-predictive control* (MPC) [15], welches hauptsächlich in der Verfahrenstechnik Anwendung fand und findet. Hier ergeben sich aber ähnliche Probleme wie bei der Adaption von FIR-Filtern, da insbesondere bei Mehrgrößensystemen sehr viele Koeffizienten nötig sind. Durch eine Ableitung des Regelgesetzes aus einer Sprungantwort ergibt sich das DMC (*dynamic matrix control*, CUTLER und RAMAKTER). Weite Verbreitung findet der *general predictive controller* (GPC), der auf AR(MA)X Modellen¹ basiert, also auf IIR-basierten Systemmodellen mit ggf. Rauschanteil.

Es existiert heute eine fast unüberschaubare Vielzahl an abgeleiteten Regelstrategien, die jedoch alle auf der oben beschriebenen Theorie basieren. Eine sehr umfassende Übersicht über viele Strategien bieten QIN und BADGWELL in [16] oder ist auch in [14] zu finden.

2.3 Auswahl eines geeigneten Verfahrens

Es wurden im vorigen Kapitel einige Strategien vorgestellt, die allgemein geeignet erscheinen, eine Dynamikkompensation schwingungsfähiger Systeme zu ermöglichen. Aufbauend auf die in Kapitel 1.2 genannten Anforderungen wurden Kriterien definiert, denen die aufgeführten Strategien entweder *sehr gut* (++), *gut* (+) oder *unzureichend* (-), bzw. *schlecht* (--) entsprechen konnten. Dadurch konnte ein qualitativer Vergleich zwischen den Algorithmen vorgenommen und eine Strategie ausgewählt werden. Die Kriterien waren:

1. Erreichbarkeit zeitlicher und örtlicher Genauigkeit
2. Anforderungen an die Trajektorie (Glattheit etc.)
3. Stabilitätsverhalten
4. Zukunftsfähigkeit (Berücksichtigung vorhandenen Zukunftswerte bzw. Bereitstellung zukünftiger Stellgrößen)
5. Implementierbarkeit (insb. Rechenzeitbeschränkungen)
6. Intuitivität (Komplexität, Verständlichkeit)

¹ Auto-regressive (Moving Average) Model with eXogenous inputs

In Tabelle 1 sind die vorgestellten Algorithmen mit den Bewertungen aufgeführt. Ist jedem + die Zahl 1 zugeordnet und entsprechend jedem – eine -1, so ergibt sich in der letzten Zeile die Summe aus allen Zeilen jeder Spalte. Der Algorithmus mit der höchsten Punktezahl scheint somit am geeignetsten.

Kriterium	Methode				
	Inverse Dynamik	Approx. der Inversen	nichtkausale Invertierung	ZPETC	Prädiktive Steuerung
1.	++	+	++	–	+
2.	0	0	–	0	+
3.	(– –)	$(+ +)^a (-)^b$	+	+	+
4.	–	–	–	–	++
5.	+	$(- -)^a (+)^b$	--	–	+
6.	++	++	--	0	+
Summe:	+ 2	+ 2	- 3	- 2	+ 7

Tabelle 1: Bewertungsschema der verschiedenen Steuerungsalgorithmen

Da die direkte Inverse nicht für nicht-minimalphasige Systeme anwendbar ist, fließt sie nicht in die Auswahl mit ein und ist nur der Vollständigkeit halber aufgeführt. Ebenso wird das Inputshaping nicht mehr aufgeführt, da es für diese Arbeit per se nicht in Frage kommt.

Die Approximation der Inversen Strecke, insbesondere mit Adaptionmechanismus, ist ein vielversprechender Ansatz. Es ergeben sich jedoch insbesondere dann Nachteile, wenn dazu eine FIR-Modellrepräsentation (dargestellt durch ein hochgestelltes a) angesetzt wird. Diese gewährleisten zwar immer Stabilität, jedoch wird durch die sehr hohe Anzahl an nötigen Koeffizienten die Rechenleistung und der Speicherbedarf drastisch erhöht. Wählt man eine andere Modellierungsgrundlage, z.B. einen IIR-Filter (Fall b), so kann die Rechenzeitanforderung stark gesenkt werden. Jedoch sind diese nicht inhärent stabil, wodurch sich Stabilitätsprobleme ergeben können.

Für die nicht-kausale Invertierung lässt sich Stabilität nachweisen und sehr gutes Folgeverhalten erzielen. Jedoch ist der implementatorische Aufwand sehr hoch, und die Theorie ist komplex. Es werden zwar Zukunftswerte der Trajektorie verwendet, jedoch nur zur Stabilisierung der Trajektorie. Es werden dadurch keine zukünftigen

Stellgrößen bereitgestellt, wie für Punkt 4. relevant. Die Möglichkeit der Trajektorienkorrektur während dem Betrieb ist trotz der Verbesserungen durch [8] eingeschränkt. Zudem sind in diversen Ausführungen Glattheitsanforderungen an die Trajektorie vorhanden.

Zero-error error tracking control bietet zwar theoretisch genaues örtliches Folgeverhalten, was jedoch mit einer Totzeit (bei differenzierenden Systemen sogar einer sehr hohen Verzögerung) erkaufte werden muss. Will man dem entgegenkommen, so kann keine stationäre Genauigkeit gewährleistet werden. Beides ist für eine präzise Positionierungsaufgabe nicht akzeptabel.

Durch die vielseitigen Entwurfsfreiheitsgrade, die sich durch das Optimierungsproblem in der prädiktiven Steuerung ergeben, kann gutes Folgeverhalten im Rahmen von Toleranzen eingestellt werden. Zwar sind dazu Versuche und Feineinstellung nötig, jedoch wird diese Möglichkeit in diesem Umfang von keinem anderen Ansatz geboten. Durch Gewichtung der Stellgrößen ist eine Beschränkung dieser möglich. Dadurch können auch nicht-glatte Trajektorien eingespeist werden. Durch vorhandene Literatur und heutige Rechenmöglichkeiten ist Stabilität gewährleistet. Ein starker Vorteil der prädiktiven Regelung ist einerseits die Berücksichtigung der (vorhandenen) zukünftigen Trajektorie, andererseits die Bereitstellung einer optimalen Sollgrößenfolge in die Zukunft. Wie später zu sehen ist, ist das resultierende Regelgesetz einfach implementierbar, und das zugrunde liegende Konzept lässt sich einfach verstehen. Es existiert umfangreiche Literatur. Aus diesen Gründen wurde sich hier für die prädiktive Regelung entschieden.

3 Theoretische Grundlagen

3.1 Grundlagen der Industrieroboter

In diesem Kapitel werden der Aufbau, wichtige Koordinatensysteme und grundlegende Eigenschaften von Industrierobotern, wie sie in diesem Projekt Anwendung fanden, besprochen. Tiefer gehende Informationen sind im Anhang C sowie in der Literatur zu finden.

3.1.1 Kinematik

In Abbildung 6 ist eine abstrahierte Skizze der Roboterkonfiguration, wie sie in diesem Projekt verwendet wurde, gezeigt. Der Roboter besteht aus sechs beweglichen Gliedern, die durch die sechs Achsen **A1...6** miteinander verbunden sind. Im Ursprung **0** befinden sich die beiden Koordinatensysteme **W** („Welt“) und **R** („Ref“). **R** ist gegenüber **W** um $+90^\circ$ um die gemeinsame y -Achse verdreht.

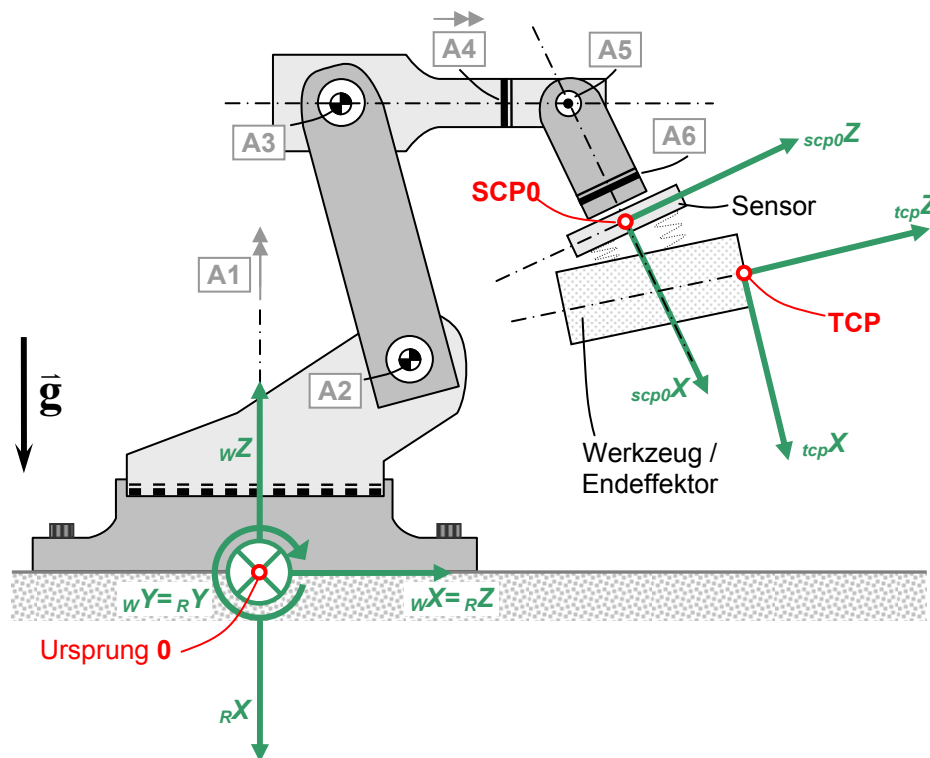


Abbildung 6: Skizze des Roboters mit ausgelenktem Endeffektor mit wichtigen Koordinatensystemen und Punkten

Am letzten Roboterglied ist der Kraft-Momenten-Sensor (kurz KMS) angeflanscht¹. Der Sensor basiert auf Stahlfedern, über deren Dehnung auf die wirkenden Kräfte und Momente zurückgeschlossen werden kann. Als „Nebenprodukt“ fällt die Relativauslenkung des Endeffektors gegenüber dem Sensor ab.

Die Spitze des Werkzeuges wird mit TCP (*tool center point*) bezeichnet. Die Solltrajektorie, die von der Trajektoriengenerierung kommt, bezieht sich stets auf ihn. In Konfiguration **K2** sitzt dort die Kamera. Das dazugehörige Koordinatensystem hat seinen Ursprung im TCP und ist körperfest mit dem Endeffektor verbunden. Mit SCP0 wird der Mittelpunkt zwischen den Federelementen bezeichnet (*spring center point*). Ihm kommt im Folgenden eine wichtige Rolle zu, denn insbesondere wird angenommen, dass er den Drehpol des Endeffektors darstellt (→ Kapitel 4.1.3). Das zugehörige Koordinatensystem ist körperfest mit dem letzten Roboterglied verbunden. Befindet sich der Endeffektor waagerechter Lage (Ausgangslage, Nulllage), so sind die Koordinatensysteme **R** und **SCP0** parallel.

Referenzierung von Punkten

Zur Referenzierung eines Punktes gibt es bei Robotern mehrere Möglichkeiten. Da die Roboterachsen allesamt als Drehachsen ausgeführt sind, lässt sich die Lage des Roboters und all seiner Glieder durch die Angabe seiner sechs Achswinkel eindeutig (außer im Fall von *kinematischen* Singularitäten, s. z. B. [17]; diese Singularitäten sind nicht mit denen der Koordinatensysteme zu verwechseln!) beschreiben. Achswerte werden mit \underline{q} bezeichnet. Um die Lage² eines Punktes auch in kartesischen Koordinaten darstellen zu können, gibt es zwei Darstellungsmöglichkeiten. Hier wird ein Punkt **B** relativ zu **A** referenziert:

- 6-dimensionale Lagevektoren ${}_K \underline{p}_{A,B}$ (K ist das Koordinatensystem)
- 4×4-dimensionale homogene Transformationsmatrizen ${}^A \underline{T}_B$ [18]

Einfach verständlich sind die Lagevektoren \underline{p} , die neben den 3 Positionen auch die 3 Orientierungen enthalten. Wie Anhang C entnommen werden kann, ergeben sich hierbei jedoch Darstellungsprobleme. Die zweite Möglichkeit ist daher die Repräsentation durch homogene Transformationsmatrizen \underline{T} , welche singularitätsfrei

¹ Je nach Konfiguration sind ein (**K1**) oder zwei (**K2**, seriell) KMS verbaut. Es ist jedoch aus Gründen der Einheitlichkeit immer die Rede von *einem* Sensor.

² Mit „Lage“ sind Position *und* Orientierung eines Körpers gemeint. Ist von der Lage *eines Punktes* die Rede, so bezieht sich die Orientierung stets auf den Körper, auf dem der Punkt körperfest liegt.

³ Ist A der Nullpunkt (also $A=0$), so kann der erste Index weggelassen werden: $p_B \equiv p_{0,B}$

ist. Die genaue Notation alle drei Darstellungsformen in dem Robotikanhang in Anhang C zu entnehmen. Implementatorisch ergibt sich folgende Kette zum Wechseln zwischen den verschiedenen Darstellungsformen:

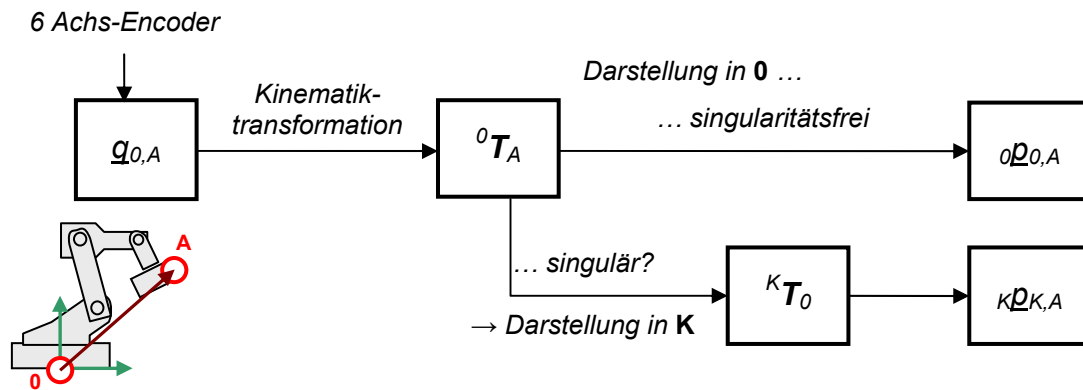


Abbildung 7: Kette der Transformationen zwischen den verschiedenen Darstellungsformen. Alle Pfeile können auch umgekehrt werden.

3.1.2 Interne Regelstruktur

In dem Robotersystem existieren eine Vielzahl an Steuerungen, Regelungen und anderen steuerungsrelevanten Komponenten nebeneinander. Einerseits existieren roboter-intern Vorsteuerungen und Regelungen, die das dynamische Verhalten des Roboters kompensieren sollen. Andererseits sind für die Montagevorhaben eigene Regelungen wie z.B. die Trajektoriengenerierung aus den Kameradaten und die Kompensation der Endeffektordynamik (diese Arbeit) nötig. Der Signalfussplan in Anlehnung an [19] sieht wie folgt aus:

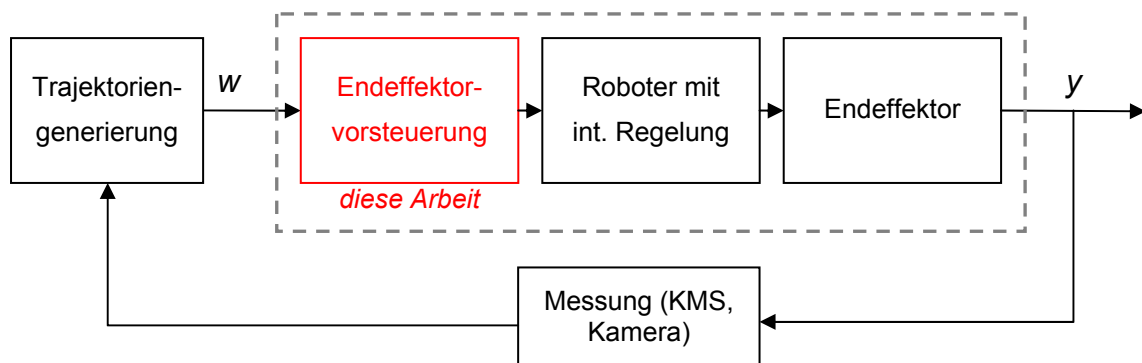


Abbildung 8: Struktur der Roboterregelung und interne Lage der Endeffektorvorsteuerung

Die Vorsteuerung, wie sie in dieser Arbeit entwickelt wird, befindet sich also direkt nach der Trajektoriengenerierung bzw. -planung. Die Sollgröße w soll so modifiziert in den Roboter geleitet werden, dass der Ausgang y möglichst gut mit w übereinstimmt. Es ist üblich, Begrifflichkeiten wie „Endeffektorvorsteuerung“ mit mathematischen Kürzeln abzukürzen. In folgender Tabelle findet sich eine Übersicht über die verwendeten Abkürzungen:

Abkürzung	bezeichnete Systemkomponente
$\Gamma(z)$	dynamische Endeffektorvorsteuerung
$\Sigma(z)$	Gesamtstrecke (<i>hier</i> bestehend aus Roboter und Endeffektor)
$R(z)$	Roboter
$E(z)$	Endeffektor
$H(z)$	Modell der Strecke $\Sigma(z)$

Tabelle 2: Bezeichnungen der Systemkomponenten

3.2 Prädiktive Steuerung

In Kapitel 2.3 wurde die Steuerungsstrategie für die Bewältigung der Aufgabenstellung gewählt und die Wahl begründet. In diesem Kapitel wird nun detailliert auf die Eigenschaften der prädiktiven Steuerung eingegangen, die Theorie hergeleitet und die Einflüsse der Parameter untersucht.

3.2.1 Modelansatz

Es gibt verschiedene Möglichkeiten, ein prädiktives Regelgesetz herzuleiten. Je nach dem, wie das Systemmodell vorliegt, lässt sich eine andere Herleitungsform benutzen. Für diskrete schwingungsfähige Systeme mit M Eingängen und N Ausgängen eignet sich eine Repräsentation der Strecke durch ein ARX-Modell:

$$\mathbf{A}(z) * \underline{y}[k] = \mathbf{B}(z) * \underline{u}[k] \cdot z^{-h} + \underline{e}[k] \quad (5)$$

Dabei steht $\mathbf{A}(z)$ für die $N \times N$ -Matrix der Polynome der Ausgangsgewichtungen und $\underline{y}[k]$ enthält die N Ausgangssignale. Dementsprechend ist $\mathbf{B}(z)$ die $N \times M$ -Matrix der

Polynome der Eingangsseite, dessen M Signale in $\underline{u}[k]$ zusammengefasst sind. z ist der diskrete Zeitverschiebungsoperator. Liegt ein totzeit-behaftetes System vor, so ist der Eingang um h Zeitschritte gegenüber dem aktuellen Zeitschritt verzögert. Allgemein lässt sich noch ein (meist unbekanntes) Fehlersignal $\underline{e}[k]$ hinzufügen.

Zur Herleitung des Regelgesetzes ist es von Vorteil, obige Gleichung durch eine Summendarstellung wiederzugeben. Hierbei wird angenommen, dass $\mathbf{A}(z)$ Diagonalcharakter¹ hat. Für die Herleitung wird die Totzeit auf $h=1$ festgelegt, da es sich bei den hier behandelten Systemen um nicht-sprungfähige² Systeme handelt. Für Systeme mit der gleichen Anzahl von M Ein- und Ausgängen ergibt sich für den n -ten Ausgang folgende Gleichung:

$$\sum_{i=0}^{\infty} a_{n,i} \cdot y_n[k-i] = \sum_{m=1}^M \sum_{i=h}^{\infty} b_{m \rightarrow n,i} \cdot u_m[k-i] \quad (6)$$

Die Multiplikationen der Signale mit den Polynommatrizen des diskreten Operators z wurden hier durch Faltungssummen ersetzt. Um den n -ten Systemausgang für einen Zeitpunkt j Schritte in der Zukunft auszurechnen, muss k durch $k+j$ ersetzt werden.

$$\sum_{i=0}^{\infty} a_{n,i} \cdot y_n[k+j-i] = \sum_{m=1}^M \sum_{i=1}^{\infty} b_{m \rightarrow n,i} \cdot u_m[k+j-i] \quad (7)$$

Die Summen rechts bzw. links lassen sich nun so aufteilen, dass die jeweiligen Terme nur in der Vergangenheit liegende, bzw. zukünftige Signalanteile enthalten:

$$\underbrace{\sum_{i=0}^{j-1} a_{n,i} \cdot y_n[k+j-i]}_{(a) \text{ zukünftige Ausgänge}} + \underbrace{\sum_{i=j}^{N_a-1} a_{n,i} \cdot y_n[k+j-i]}_{(b) \text{ vergangene Ausgänge}} = \underbrace{\sum_{m=1}^M \sum_{i=1}^j b_{m \rightarrow n,i} \cdot u_m[k+j-i]}_{(c) \text{ zukünftige Eingänge}} + \underbrace{\sum_{m=1}^M \sum_{i=j+1}^{N_b-1} b_{m \rightarrow n,i} \cdot u_m[k+j-i]}_{(d) \text{ vergangene Eingänge}} \quad (8)$$

¹ Für viele Schwingungssysteme ist diese Annahme gerechtfertigt. Diese Aussage ist gleichbedeutend mit der Idee, dass der schwingungsfähige Ausgang von allen (gekoppelten) Eingängen angeregt werden kann, jedoch dass sich die Ausgänge untereinander nicht beeinflussen (\rightarrow Kapitel 4.1.3.2).

² Dies bedeutet, dass der aktuelle Systemausgang nicht unmittelbar vom Eingang zur gleichen Zeit abhängen kann, sondern min. einen Zeitschritt braucht, um auf den Ausgang zu wirken.

Besonders die Summationsgrenzen sind zu beachten. Es wurde angenommen, dass jeder Ausgang N_a a -Parameter und jeder Eingang N_b b -Parameter hat, nachdem diese Summen abgebrochen werden. Nachfolgende Parameter sind per Definition 0.

Man kann nun diese Gleichung für alle $n=m=1\dots M$ Freiheitsgrade für zukünftige Ausgänge von $j=1\dots N_p$ in Matrixschreibweise wie folgt zusammenfassen:

$$\mathcal{A}_+ \cdot \underline{Y}_+ + \mathcal{A}_- \cdot \underline{Y}_- = \mathcal{B}_+ \cdot \underline{U}_+ + \mathcal{B}_- \cdot \underline{U}_- \quad (9)$$

Auf diesen Schritt soll hier nicht weiter eingegangen werden. Die Struktur der Matrizen kann in Anhang A nachgesehen werden. Die Struktur der Signalterme ist wie folgt:

$$\begin{aligned} \underline{Y}_+ &= \begin{bmatrix} y^T[k+1] & \cdots & y^T[k+(N_p-1)] & y^T[k+N_p] \end{bmatrix}^T \\ \underline{Y}_- &= \begin{bmatrix} y^T[k] & \cdots & y^T[k-(N_a-2)] & y^T[k-(N_a-1)] \end{bmatrix}^T \\ \underline{U}_+ &= \begin{bmatrix} u^T[k] & \cdots & u^T[k+(N_p-2)] & u^T[k+(N_p-1)] \end{bmatrix}^T \\ \underline{U}_- &= \begin{bmatrix} u^T[k-1] & \cdots & u^T[k-(N_b-2)] & u^T[k-(N_b-1)] \end{bmatrix}^T \end{aligned} \quad (10)$$

Die Großschreibung soll verdeutlichen, dass es sich bei diesen Vektoren um Vektoren von Vektoren handelt – wie in Gleichung (10) dargestellt, besteht bspw. \underline{Y}_+ aus allen zukünftigen N_p Signalen

$$\underline{y}[k+j] = \begin{bmatrix} \vdots \\ y_n[k+j] \\ \vdots \end{bmatrix} \in \mathcal{R}^{M \times 1} \text{ mit } n=1\dots M \quad (11)$$

Um nun den Vektor der zukünftigen, also prädizierten Systemausgänge zu erhalten, muss Gleichung (9) nach \underline{Y}_+ aufgelöst werden:

$$\begin{aligned} \underline{Y}_+ &= \mathcal{A}_+^{-1} \cdot \mathcal{B}_+ \cdot \underline{U}_+ + \mathcal{A}_+^{-1} \cdot \mathcal{B}_- \cdot \underline{U}_- - \mathcal{A}_+^{-1} \cdot \mathcal{A}_- \cdot \underline{Y}_- \\ &= \mathcal{C}_+ \cdot \underline{U}_+ + \mathcal{C}_- \cdot \underline{U}_- - \mathcal{D} \cdot \underline{Y}_- \end{aligned} \quad (12)$$

\mathcal{A}_+ besitzt eine untere Dreiecksstruktur und ist somit immer invertierbar, solange alle Diagonalelemente (d.h. alle a_0 -Parameter) ungleich 0 sind. Dies ist für unsere Systeme, in denen $a_0=1$ ist, immer erfüllt.

3.2.2 Optimale Regelung

Die modellbasierte prädiktive Steuerung basiert auf der Minimierung der zukünftigen Abweichung des Systemausgangs zu dem Sollgrößenverlauf. Bezeichnet man den zukünftigen Sollgrößenverlauf mit \underline{W}_+ und den zukünftigen Systemausgang mit \underline{Y}_+ , so lässt sich der prädizierte Fehler \underline{E}_+ wie folgt angeben:

$$\underline{E}_+ = \underline{W}_+ - \underline{Y}_+ = f(\underline{U}_+) \quad (13)$$

Dazu muss natürlich der Sollgrößenverlauf möglichst bis N_p Zeitschritte in der Zukunft bekannt sein. Ist er dies nicht, so können ggf. fehlende Werte aus alten Werten extrapoliert werden.

In Anlehnung an die optimale Regelung, in der ein skalares Gütemaß \mathcal{V} , das kritische Eigenschaften des Systems bzw. der Regelung umfasst, minimiert wird, so wird auch hier versucht, den zukünftigen Fehler aus Gleichung (13) mit mathematischen Hilfsmitteln zu minimieren. Dazu wird anfangs ein geeignetes Maß für die Güte der Regelung festgelegt (hier eine Funktion in Abhängigkeit des prädizierten Fehlers, der wiederum von den zukünftigen Stellgrößen abhängt, s. Gleichung (13) und dann dieses Maß mathematisch nach den zukünftigen Stellgrößen abgeleitet. Die resultierende Gleichung wird den üblichen Regeln einer Minimierung entsprechend zu Null gesetzt und nach den zukünftigen Stellgrößen \underline{U}_+ aufgelöst. Anders gesagt: Es wird auf mathematische Weise ein \underline{U}_+ so gewählt, dass das Gütemaß für den betrachteten Zeitraum minimal wird.

Verschiedene Gütemaße sind aus der optimalen Regelung bekannt (s. z. B. [20]); mathematisch einfach und weit verbreitet ist ein *quadratisches* Gütemaß \mathcal{V}_Q mit diagonalen¹ Gewichtungs- oder auch Strafmatrix \mathbf{Q} . Je größer der einem Freiheitsgrad n zu einem Zeitpunkt k zugeschriebene Wert q_n^k , desto „stärker“ wird versucht, den zukünftigen Fehler dieses Freiheitsgrades zu jenem zukünftigen Zeitpunkt zu minimieren. Der Fehler wird *bestraft*.

$$\mathcal{V}_Q = \underline{E}_+^T \cdot \mathbf{Q} \cdot \underline{E}_+ \in \mathcal{R}^1 \quad (14)$$

Würde man Gleichung (14) nun direkt nach \underline{U}_+ ableiten, erhielte man einen Ausdruck für \underline{U}_+ , welcher die zukünftigen Fehler *unter allen Umständen* minimieren würde. Bei schwach gedämpften oder gar nicht-minimalphasigen Systemen kann dies zu über

¹ Physikalisch sinnvoll deutbar sind i.A. nur diagonale Gewichtungsmatrizen

alle Grenzen wachsenden Stellgrößenausschlägen und Instabilität führen. Wie später noch zu sehen ist, beinhaltet ein solches prädiktives Regelgesetz eine Inverse der Modelldynamik und kann somit leicht instabil werden (es ist *invers-artig*). Es ist daher nötig, das Gütefunktional um einen weiteren Strafterm für die Stellgrößen zu erweitern. Dieser soll bewirken, dass das Regelziel nicht mit beliebig großen, sondern ebenso einer Minimierung unterworfenen Stellgrößen erreicht wird. Die Einflüsse davon werden später noch behandelt. Eine in der Literatur übliche Form sieht wie folgt aus:

$$\mathcal{V}_{QR} = \underline{E}_+^T \cdot \mathbf{Q} \cdot \underline{E}_+ + \underline{U}_+^T \cdot \mathbf{R} \cdot \underline{U}_+ \quad (15)$$

Das \mathbf{R} erfüllt die gleiche Aufgabe wie das \mathbf{Q} ; es dient zur unterschiedlichen zeitlichen und die Freiheitsgrade betreffende Gewichtung der Stellgrößen. Die Dimensionen der beiden Matrizen sind wie folgt:

$$\begin{aligned} \mathbf{Q} &\in \mathcal{R}^{N_p \times N_p} \\ \mathbf{R} &\in \mathcal{R}^{N_c \times N_c} \end{aligned} \quad (16)$$

Damit ist deutlich, dass \underline{U}_+ und \underline{W}_+ nicht dieselbe Dimension haben müssen – es kann also weiter in die Zukunft prädiziert als gesteuert werden. Folglich bezeichnet man N_p als *Prädiktionshorizont*, N_c als *Steuerhorizont*.

Nun enthalten bei einem Robotersystem sowohl \underline{U}_+ als auch \underline{W}_+ eine Information über die kartesische Bahn eines Punktes, der mit dem Roboter verbunden ist. Eine Minimierung der Stellgrößen würde bewirken, dass die Steuerung das Regelziel so erreichen möchte, indem sie sich möglichst *nahe* am Roboternullpunkt, also dem Koordinatenursprung im Roboterfundament, bewegt. Es ist jedoch unerheblich und von der Anwendung abhängig, wo sich der Roboter absolut befindet – nicht die Stellgröße darf *absolut*, sondern nur eine *relative* Bezugsgröße darf minimiert werden.

Es gibt dazu die Möglichkeit, die *zeitliche* Änderung der Stellgröße von einem auf den anderen Zeitschritt zu bestrafen, also ein $\Delta \underline{U}_+$. Da sich jedoch \underline{U}_+ und \underline{W}_+ in diesem Anwendungsfall nur wenig unterscheiden, und vielmehr eine Überlagerung der Stellgröße über das eigentliche Referenzsignal nötig ist, wurde ein Gütemaß eingeführt, dass die Abweichung zwischen Soll- und Stellsignal minimal hält. Damit wird neben einer Stabilisierung und Begrenzung der Stellgrößen, auch eine Art „Durchgriff“ des Sollsignals erreicht.

Das resultierende Gütefunktional lautet dann:

$$\begin{aligned}\mathcal{V}_{QR} &= \underline{E}_+^T \cdot \mathbf{Q} \cdot \underline{E}_+ + \left(\underline{W}_+^c - \underline{U}_+ \right)^T \cdot \mathbf{R} \cdot \left(\underline{W}_+^c - \underline{U}_+ \right) \\ &= \left(\underline{W}_+^p - \underline{Y}_+ \right)^T \cdot \mathbf{Q} \cdot \left(\underline{W}_+^p - \underline{Y}_+ \right) + \left(\underline{W}_+^c - \underline{U}_+ \right)^T \cdot \mathbf{R} \cdot \left(\underline{W}_+^c - \underline{U}_+ \right)\end{aligned}\quad (17)$$

Zur Unterscheidung der beiden Sollgrößenverläufe wurden sie jeweils mit einem Index p bzw. c gekennzeichnet. Tritt \underline{W} ohne zusätzlichen Index auf, so ist von \underline{W}^p die Rede. Somit ist:

$$\begin{aligned}\underline{W}_+^p &= \left[\underline{w}^T[k+1] \quad \cdots \quad \underline{w}^T[k+N_p] \right]^T \in \mathcal{R}^{M \cdot N_p \times 1} \\ \underline{W}_+^c &= \left[\underline{w}^T[k+1] \quad \cdots \quad \underline{w}^T[k+N_c] \right]^T \in \mathcal{R}^{M \cdot N_c \times 1}\end{aligned}\quad (18)$$

Zur Minimierung des Gütemaßes muss dessen partielle Ableitung nach \underline{U}_+ zu null gesetzt werden:

$$\frac{\partial \mathcal{V}_{QR}}{\partial \underline{U}_+} \stackrel{!}{=} 0 \quad (19)$$

Unter Verwendung der symbolischen Ableitungsregeln für Matrizen und Vektoren ergibt sich ein Ausdruck für \underline{U}_+ , welcher das Gütemaß minimiert:

$$\begin{aligned}\underline{U}_+ &= \underbrace{\left(\mathbf{C}_+^T \cdot \mathbf{Q} \cdot \mathbf{C}_+ + \mathbf{R} \right)^{-1}}_{\stackrel{!}{=}\mathbf{K}} \cdot \left(\overbrace{\mathbf{C}_+^T \cdot \mathbf{Q} \cdot \left(\underline{W}_+^p - \underline{F}_+ \right)}^{:=\mathbf{G}_+} + \mathbf{R} \cdot \underline{W}_+^c \right) \\ &= \mathbf{K} \cdot \left(\mathbf{G}_+ \cdot \left(\underline{W}_+^p - \underline{F}_+ \right) + \underbrace{\mathbf{R} \cdot \underline{W}_+^c}_{\text{Durchgriff}} \right)\end{aligned}\quad (20)$$

Gleichung (20) ist das prädiktive Steuergesetz, welches in dieser Arbeit Anwendung findet. Sie ist eine zentrale Gleichung. In ihr wird \underline{F}_+ als die *freie Bewegung* des Systems *ohne* zukünftigen Stelleingriff bezeichnet (vgl. auch Gleichung (12)):

$$\underline{F}_+ = \mathbf{C}_- \cdot \underline{U}_- - \mathbf{D} \cdot \underline{Y}_- \quad (21)$$

Eine Übersicht über alle Dimensionen ist im Anhang A gegeben. Gleichung (20) lässt sich auch in einem Signalflussplan zusammenfassen:

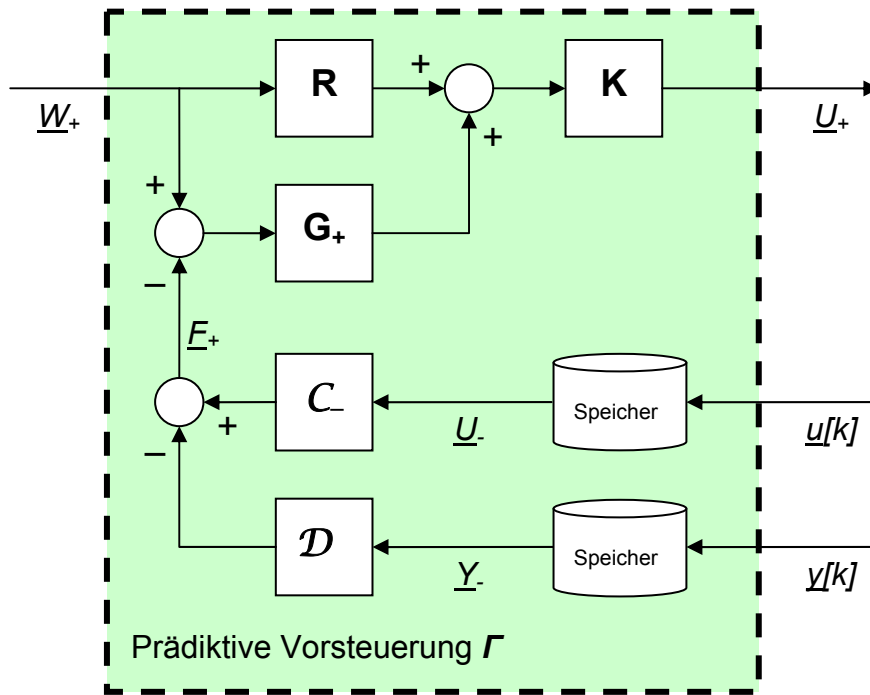


Abbildung 9: Signalflussplan des Regelgesetzes aus Gleichung (20)

3.2.3 Berechnung der freien Systembewegung

Abbildung 9 und Gleichung (21) zeigen, dass zur Berechnung der freien Systembewegung \underline{E}_+ der Vektor \underline{Y}_- benötigt wird. Dieser muss bei der Implementierung durch Zwischenspeicherung des aktuellen Vektors $\underline{y}[k]$ erstellt werden. Es gibt jedoch mehrere Möglichkeiten, $\underline{y}[k]$ einzuspeisen. Auf den ersten Blick mag naheliegen, für $\underline{y}[k]$ den Systemausgang heranzuziehen. Dies kann jedoch Nachteile mit sich bringen, die hier besprochen werden sollen. Weiterhin besteht die Möglichkeit, eine modellbasierte Prädiktion des Systemausgangs zu berechnen – anstatt des realen Systemausgangs wird dann mithilfe des Systemmodells der aktuelle reale Systemausgang „vorhergesagt“. Eine Mischform ist die „Anreicherung“ des prädizierten Signals mit Messungen am Ausgang.

Auf die Vor- und Nachteile der drei Methoden soll hier kurz eingegangen werden. Welche sich für den Anwendungsfall einer Trajektorienvorsteuerung mit Robotern unter welchen Bedingungen am besten eignet, wird in späteren Kapiteln behandelt.

3.2.3.1 Gemessener Systemausgang

Prinzipiell ist natürlich dazu notwendig, dass das Ausgangssignal $\underline{y}[k]$ überhaupt gemessen werden kann. Das ist für viele Anwendungen – so auch für diese – zwar der Fall. Im Allgemeinen kann jedoch nicht davon ausgegangen werden, dass das Ausgangssignal messbar ist. Die Struktur der gesamten Strecke unter Verwendung des gemessenen Systemausgangs ist in Abbildung 10 dargestellt. Hierbei wurde die Vorsteuerung unter der Übertragungsfunktion Γ zusammengefasst.

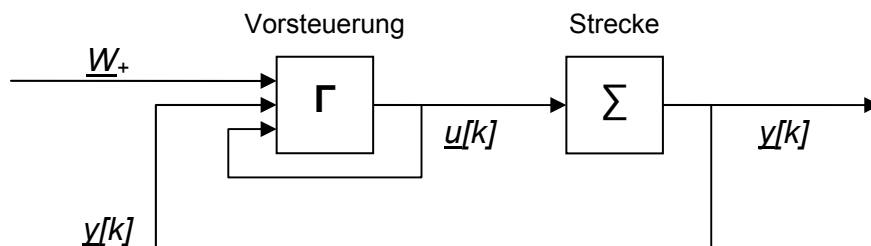


Abbildung 10: Signalflussplan mit Rückkopplung des Ausgangs

Die Rückkopplung des gemessenen Ausgangssignals führt jedoch zu einer Schließung des Regelkreises, was bei Modellfehlern unvorhersehbares Stabilitätsverhalten mit sich bringen kann. Insbesondere bei totzeit-behafteten und schwingungsfähigen Systemen müssen Systemmodell (aus dem die Vorsteuerung erwächst) und echte Strecke möglichst gut übereinstimmen. Ist dies nicht gegeben, kann selbst eine kleine Phasenabweichung zwischen dem rückgekoppelten Signal und dem „internen“ Modellausgang Instabilität hervorrufen.

Eine Rückkopplung des gemessenen Ausgangssignals hat sich für dieses schwingungsfähige System als nicht praktikabel erwiesen.

3.2.3.2 Prädiktion

Wird $\underline{y}[k]$ rein prädiktiv, also aus den vergangenen Systemeingängen, modellbasiert berechnet, so lässt sich die Stabilität der Vorsteuerung besser garantieren. Eine solche Struktur ist in Abbildung 11 dargestellt. Hierbei wird das Systemmodell H mit den Stellgrößen für das echte System gespeist – es läuft „parallel“. Anstatt des echten Systemausgangs wird nun der Ausgang des Modells verwendet.

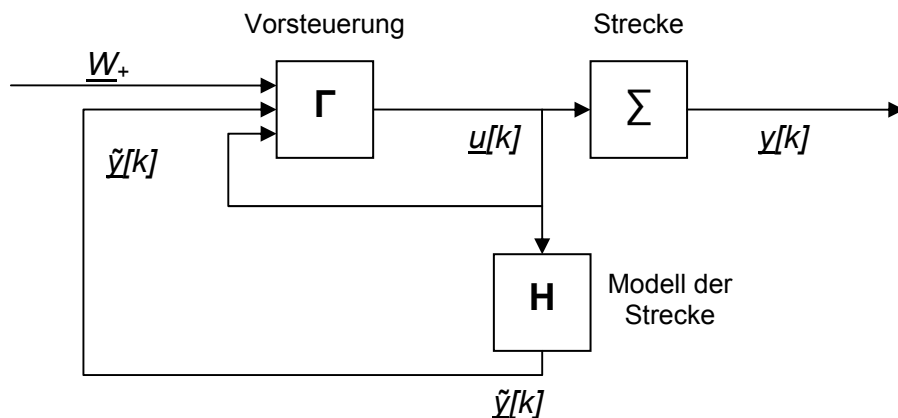


Abbildung 11: Signalflussplan mit Einspeisung eines prädizierten Systemausgangs

Dieses Verfahren hat ein einige Vorteile. Einerseits lässt sich so die Stabilität der Vorsteuerung mathematisch beweisen. Ist das System totzeit-behaftet, so kann das Modell zudem verwendet werden, den Systemausgang für Zeiten kleiner der Totzeit vorherzusagen, also zu prädizieren. Nachteilig ist, dass bei größeren Abweichungen zwischen H und Σ kein gutes Regelergebnis erzielt werden kann. Es wird keine Information über das tatsächliche Systemverhalten an die Vorsteuerung rückgemeldet. Dem muss man allerdings entgegenhalten, dass es sich hierbei um eine Steuerung und keine Regelung handelt.

Es hat sich in Experimenten gezeigt, dass diese Form der Ausgangseinspeisung eine sichere und akzeptable ist.

3.2.3.3 Angereicherte Mischform

Ebenso denkbar ist eine Mischung beider Ansätze, wie sie auch in der Literatur beschrieben wird [21]. Die Idee ist, den prädizierten Ausgang \tilde{y} um einen gemessenen Anteil „anzureichern“. Das letztendlich eingespeiste Signal \hat{y} lautet:

$$\hat{y}[k] = \tilde{y}[k] + \mathbf{P}(z) \cdot (y[k] - \tilde{y}[k]) = \mathbf{P}(z)y[k] + \underbrace{(1 - \mathbf{P}(z))}_{=0 \Leftrightarrow \mathbf{P}=1} \tilde{y}[k] \quad (22)$$

Für den Sonderfall $P(z)=1$ geht die Einspeisung in die Struktur aus Kapitel 3.2.3.1 über. Oft wird $P(z)$ zu einer einfachen Totzeit, einem Tiefpassfilter oder einem Faktor gewählt.

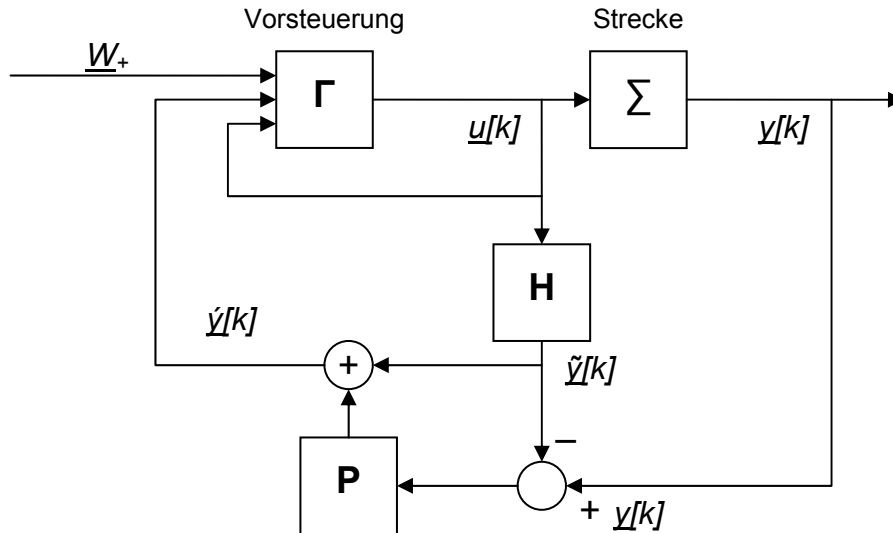


Abbildung 12: Signalflussplan mit Einspeisung eines angereicherten prädizierten Systemausgangs

Da sich dies also ebenso nur für schwach-dynamische Systeme bewährt hat, soll hier nicht im Detail darauf eingegangen werden.

3.2.4 Übertragungsfunktion und stationäre Genauigkeit

Ein Nachteil des bisherigen Regelgesetzes ist, dass es im Allgemeinen keine stationäre Genauigkeit gewährleistet. Insbesondere die „künstlich“ eingebrachten Gewichtungsmatrizen \mathbf{Q} und \mathbf{R} lassen den Übertragungsfaktor von $\mathbf{1}$ abweichen. Um stationäre Genauigkeit zu erreichen, muss die stationäre Übertragungsmatrix $\mathbf{\Gamma}_{stat}$ der Übertragungsfunktion $\mathbf{\Gamma}(z)$ der Vorsteuerung gefunden werden. Im ersten Schritt wird dazu die Übertragungsfunktion hergeleitet.

Da generell nur das aktuellste Signal $\underline{u}[k]$ aus \underline{U}_+ (d.h., die ersten M Zeilen) an das System gesendet werden kann, darf auch nur dieses zur Herleitung der Übertragungsfunktion verwendet werden.

$$\underline{u}[k] = \mathbf{k} \cdot \left(\mathbf{G}_+ \cdot \left(\underline{W}_+^p - \mathbf{C}_- \cdot \underline{U}_- + \mathbf{D} \cdot \underline{Y}_- \right) + \mathbf{R} \cdot \underline{W}_+^c \right) \quad (23)$$

Hierbei enthält \mathbf{k} die ersten M Zeilen von \mathbf{K} . Transformiert man Gleichung (23) mithilfe der z-Transformation vom diskreten Zeit- in den diskreten Bildbereich, so ergibt sich folgender Ausdruck für $\underline{\mathbf{U}}(z)$:

$$\underline{\mathbf{U}}(z) = \left(\underbrace{\left(\mathbf{1} - \mathbf{k}\mathbf{G}_+ \cdot \left(\mathcal{D}\zeta_{-(Na-1)}^{-0} \mathbf{H}(z) - \mathbf{C}_- \zeta_{-(Nb-1)}^{-1} \right) \right)^{-1} \cdot \mathbf{k} \cdot \left(\mathbf{G}_+ \zeta_{+Np}^{+1} + \mathbf{R}\zeta_{+Nc}^{+1} \right)}_{\text{Übertragungsfunktion } \Gamma(z)} \right) \cdot \underline{\mathbf{W}}(z) \quad (24)$$

Hier bei wurde der Fall der Prädiktion (Kapitel 3.2.3.2) mit $\underline{\mathbf{y}}(z) = \mathbf{H}(z) \cdot \underline{\mathbf{u}}(z)$, $\mathbf{H}(z)$ das Systemmodell, angenommen. Der mehrdimensionale Zeitverschiebeoperator ζ wird im mathematischen Anhang erläutert. Ähnlich wie der eindimensionale Verschiebeoperator geht ζ für $k \rightarrow \infty$ in eine Form der Einheitsmatrix, hier mit $M\check{\mathbf{I}}_{|o-u|}$ bezeichnet, über:

$$\zeta_u^o = \begin{bmatrix} \vdots \\ \mathbf{1}_{M \times M} \cdot z^j \\ \vdots \end{bmatrix} \xrightarrow[z^j \rightarrow 1]{k \rightarrow \infty} \begin{bmatrix} \vdots \\ \mathbf{1}_{M \times M} \\ \vdots \end{bmatrix} := M\check{\mathbf{I}}_{|o-u|} \in \mathcal{R}^{M|o-u| \times M} \quad (25)$$

mit $j = o \dots u \in \mathbb{Z}$

$M\check{\mathbf{I}}_{|o-u|}$ ist damit eine $|o-u|$ -fache Übereinanderstellung von $M \times M$ -Einheitsmatrizen. M ist im Folgenden aus Gründen der Übersichtlichkeit nicht als Argument des Operators dargestellt. Der stationäre Übertragungsfaktor der Übertragungsfunktion

$$\Gamma(z) = \left(\mathbf{1} - \mathbf{k}\mathbf{G}_+ \cdot \left(\mathcal{D}\zeta_{-(Na-1)}^{-0} \mathbf{H}(z) - \mathbf{C}_- \zeta_{-(Nb-1)}^{-1} \right) \right)^{-1} \cdot \mathbf{k} \cdot \left(\mathbf{G}_+ \zeta_{+Np}^{+1} + \mathbf{R}\zeta_{+Nc}^{+1} \right) \quad (26)$$

nach Gleichung (24) ist somit also

$$\Gamma(1) := \Gamma_{stat} = \left(\mathbf{1} - \mathbf{k}\mathbf{G}_+ \cdot \left(\mathcal{D}\check{\mathbf{I}}_{-(Na-1)}^{-0} \mathbf{H}(1) - \mathbf{C}_- \check{\mathbf{I}}_{-(Nb-1)}^{-1} \right) \right)^{-1} \cdot \mathbf{k} \cdot \left(\mathbf{G}_+ \check{\mathbf{I}}_{+Np}^{+1} + \mathbf{R}\check{\mathbf{I}}_{+Nc}^{+1} \right) \quad (27)$$

worin $\mathbf{H}(1) = \mathbf{H}_{stat}$ der statische Übertragungsfaktor des Systemmodells ist. Damit der gesamte Übertragungsfaktor von der Sollgröße $\underline{\mathbf{w}}$ auf den Ausgang $\underline{\mathbf{y}}$ 1 ist, muss auch die Strecke in der Berechnung des Gesamtübertragungsfaktors \mathbf{F} berücksichtigt werden.

Der Gesamtübertragungsfaktor ergibt sich damit zu

$$\mathbf{F} = \mathbf{\Gamma}(1) \cdot \mathbf{\Sigma}(1) = \mathbf{\Gamma}_{stat} \cdot \mathbf{\Sigma}_{stat} \quad (28)$$

Da die Strecke nie hinreichend bekannt sein wird, muss jedoch die Annahme getroffen werden, dass $\mathbf{\Sigma}_{stat}$ mit dem Übertragungsfaktor des Systemmodells, \mathbf{H}_{stat} , identisch ist. Es ergibt sich also:

$$\mathbf{F} = \mathbf{\Gamma}_{stat} \cdot \mathbf{H}_{stat} \in \mathcal{R}^{M \times M} \quad (29)$$

Zur Gewährleistung stationärer Genauigkeit des Gesamtsystems müsste nun die Sollgröße mit der Inversen von \mathbf{F} skaliert an das System gesendet werden. Nun hat \underline{W}_+ jedoch die Dimension $M \cdot N_p \times 1$, womit jeder zukünftige Eintrag von \underline{W}_+ mit der Korrekturmatrix skaliert werden muss. Es ergibt sich eine Gesamtkorrekturmatrix $\mathbf{\Phi}$, die aus N_p diagonal angeordneten Matrizen \mathbf{F} besteht.

$$\underline{W}_+ \rightarrow \begin{bmatrix} \ddots & & \mathbf{0} \\ & \mathbf{F}^{-1} & \\ \mathbf{0} & & \ddots \end{bmatrix} \cdot \underline{W}_+ := \mathbf{\Phi} \cdot \underline{W}_+ \quad \text{und} \quad \mathbf{\Phi} = \text{diag}_{N_p} \{ \mathbf{F}^{-1} \} \quad (30)$$

3.2.5 Stabilität

Kurz soll auf den theoretischen Nachweis der Stabilität eingegangen werden. Aus Gleichung (26) lassen sich mit geeigneten mathematischen Hilfsmitteln die Pole der Übertragungsfunktion $\mathbf{\Gamma}(z)$ errechnen. Beispielsweise lässt sich die Übertragungsfunktion in MATLAB© mit dem Befehl

```
Gamma = tf(Z, N, t)
```

mit Z den Zähler- und N den Nennerkoeffizienten, sowie t der Abtastzeit τ erstellen und deren Pole numerisch mit

```
pole(Gamma)
```

berechnen.

Graphisch können die Pole mit

pzmap (Gamma)

dargestellt werden. Es gilt das Stabilitätskriterium für diskrete Systeme, wonach alle Pole des Systems strikt innerhalb des Einheitskreises liegen müssen. Es gibt auch andere Stabilitätskriterien, wie z.B. LYAPUNOV-Funktionen. Diese werden u.a. in [22] behandelt. Die Stabilität wird maßgeblich durch die Wahl der Parameter, wie im folgenden Kapitel besprochen, beeinflusst.

3.2.6 Einflüsse der Parameter

Wie bei vielen Regelungsansätzen besteht eine zentrale Aufgabe in der Wahl der einstellbaren Regelparameter. Beim hier beschriebenen prädiktiven Regler sind dies in erster Linie folgende Einflussfaktoren:

1. die Strafmatrizen \mathbf{Q} und \mathbf{R}
2. der Prädiktionshorizont N_p
3. der Steuerungshorizont N_c

Es werden Richtlinien, Deutungen und Daumenregeln für die Wahl der Parameter bzw. das Verhalten des Reglers bei Parameteränderung angegeben.

3.2.6.1 Fehlerstrafmatrix \mathbf{Q} und Stellgrößenstrafmatrix \mathbf{R}

Die Matrizen \mathbf{Q} und \mathbf{R} dienen dazu, Fehler bzw. Stellgrößenausschläge in einzelnen Freiheitsgraden zu zukünftigen Zeitpunkten unterschiedlich stark zu gewichten oder auch zu „bestrafen“. Beide Gewichtungsmatrizen haben Diagonalstruktur. Sie haben folgende Struktur (exemplarisch für die Matrix \mathbf{Q} gezeigt):

$$\mathbf{Q} = \begin{bmatrix} \mathbf{q}^1 & & & \mathbf{0} \\ & \ddots & & \\ & & \mathbf{q}^k & \\ & & & \ddots \\ \mathbf{0} & & & & \mathbf{q}^{N_p} \end{bmatrix} \quad \text{mit } \mathbf{q}^k = \begin{bmatrix} q_1^k & & 0 \\ & \ddots & \\ 0 & & q_M^k \end{bmatrix} > 0 \quad (31)$$

Die Fehlerstrafmatrix \mathbf{Q} besteht also aus $k=1 \dots N_p$ positiv-definiten Einzelmatrizen \mathbf{q}^k . Jede Untermatrix legt dabei die Gewichtung des prädizierten Fehlers \underline{e}_+ für alle M

Freiheitsgrade zu einem Zeitpunkt k fest. Entsprechend gilt: \mathbf{R} besteht aus $k=1 \dots N_c$ Einzelmatrizen \mathbf{r}^k , die Stellgrößenausschläge zum Zeitpunkt k in der Zukunft „bestraft“. Die beiden Matrizen stehen in Konkurrenz – werden die q_n^k betragsmäßig vergrößert, versucht die Steuerung, Fehler des Freiheitsgrades n zu minimieren. Es entstehen dementsprechend in den mit n verkoppelten Freiheitsgraden größere Stellbewegungen. Vergrößert man hingegen die r_n^k in ähnlichem oder gleichem Maße, so wird wiederum versucht, die Stellgrößenausschläge zu minimieren. Die Effekte kompensieren sich – es kommt auf das Verhältnis von \mathbf{Q} zu \mathbf{R} an.

Maßnahmen				Effekte
q_n	in k	r_n	in k	
\uparrow / \downarrow	–	\leftrightarrow	–	Fehler in Freiheitsgrad n wird durch stärkere/schwächere Stellgrößenausschläge in n und seinen gekoppelten Freiheitsgraden verringert/erhöht
\leftrightarrow	–	\uparrow / \downarrow	–	Stellgrößenausschläge in n werden verstärkt/vermindert. Fehler in n und seinen gekoppelten FHGen verringern/erhöhen sich.
\leftrightarrow	\nearrow	\leftrightarrow	–	Steuerung reagiert tendenziell früher auf zukünftige Sollgrößenänderungen

Tabelle 3: Einflüsse der Strafmatrizen auf die Stellgrößen

Tendenzen zur Wahl der Koeffizienten von \mathbf{Q} zu \mathbf{R} sind in Tabelle 3 gegeben. Ausgangspunkt sind beliebig gewählte Koeffizienten q_n^k und r_n^k für einen exemplarischen Freiheitsgrad n , die nach und nach wie gezeigt verändert werden. Hierbei steht ein \uparrow für eine Erhöhung, \downarrow für eine Verringerung des neuen gegenüber des alten Wertes. Mit \leftrightarrow wird ein Gleichbleiben angedeutet. Die Spalten „in k “ bedeuten, ob die Koeffizienten innerhalb der Matrix von ihrem Anfangswert $(\cdot)_n^1$ ansteigen (\nearrow), absteigen (\searrow) oder konstant (–) bleiben. Die Liste ist unvollständig und soll einen Anhaltspunkt geben.

Für schwach-gedämpfte Freiheitsgrade sollte q_n eher größer als r_n sein, damit die Schwingungen ausgedämpft werden können. Alternativ kann auch versucht werden, n durch die Verringerung der r_n s gekoppelter Freiheitsgrade zu dämpfen. Die Matrix \mathbf{R} kann zur Stabilisierung der Vorsteuerung dienen. Betrachtet man Gleichung (20), so ist einfach einzusehen, dass eine „Vergrößerung“ von \mathbf{R} durch die Inversion zu einer betragsmäßigen Verkleinerung von \mathbf{K} führt. Ein zukünftiger Fehler, möge er noch so groß sein, wird weniger bestraft. Natürlich ist eine Konsequenz, dass

dadurch das Steuerungsergebnis verschlechtert wird. Insbesondere kann gezeigt werden, dass für ein ausreichend großes \mathbf{R} gilt:

$$\mathbf{K} \xrightarrow{\det(\mathbf{R}) \uparrow} \mathbf{R}^{-1} \quad (32)$$

➔ Eine Erhöhung des Betrags $|\mathbf{R}|$ führt tendenziell zu einer Stabilisierung

Es soll noch kurz auf die Notation eingegangen werden. Da sich die einzelnen Elemente von \mathbf{q}^k bzw. \mathbf{r}^k in erster Linie in ihrer Größenordnung unterscheiden, wird hier auch folgende Notation verwendet:

$$\mathbf{q}^k = \varepsilon \cdot \underset{\Lambda}{diag}\{10^\Lambda\} \longrightarrow \mathbf{q}^k = \varepsilon \cdot 10^{diag\{\Lambda\}} \quad (33)$$

ε ist der Skalierungsfaktor, Λ ist die Menge der M Exponenten; Details s. Anhang B.

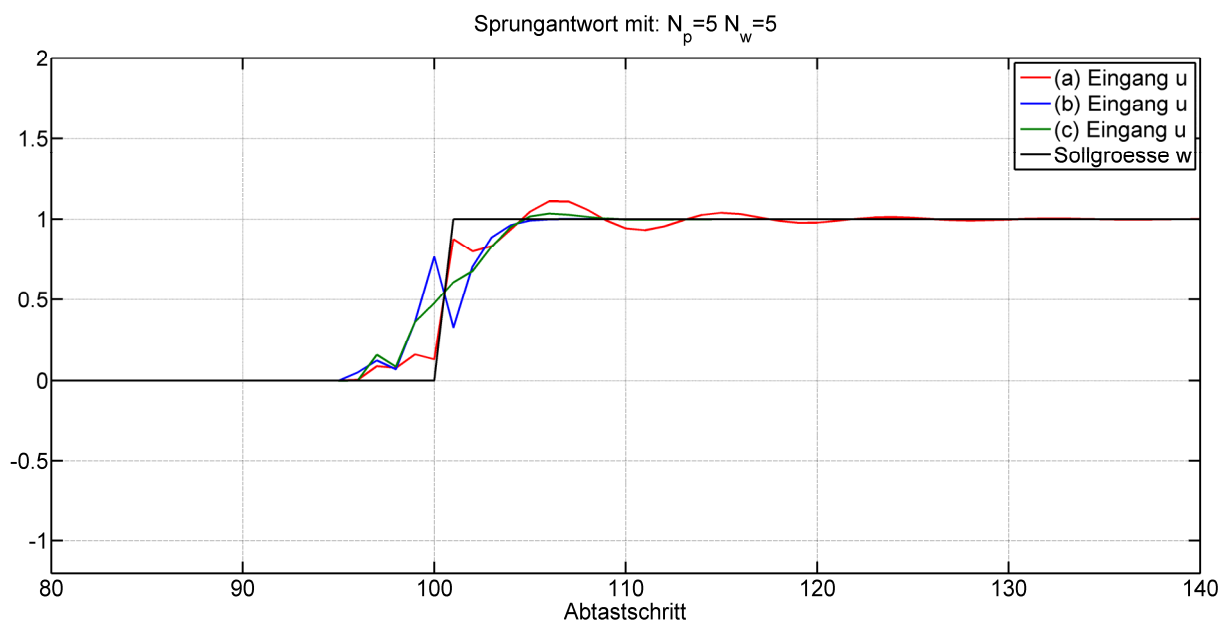


Abbildung 13: Stellgrößen für einen Sprung in einem schwach-gedämpften System für drei verschiedene Parametersätze

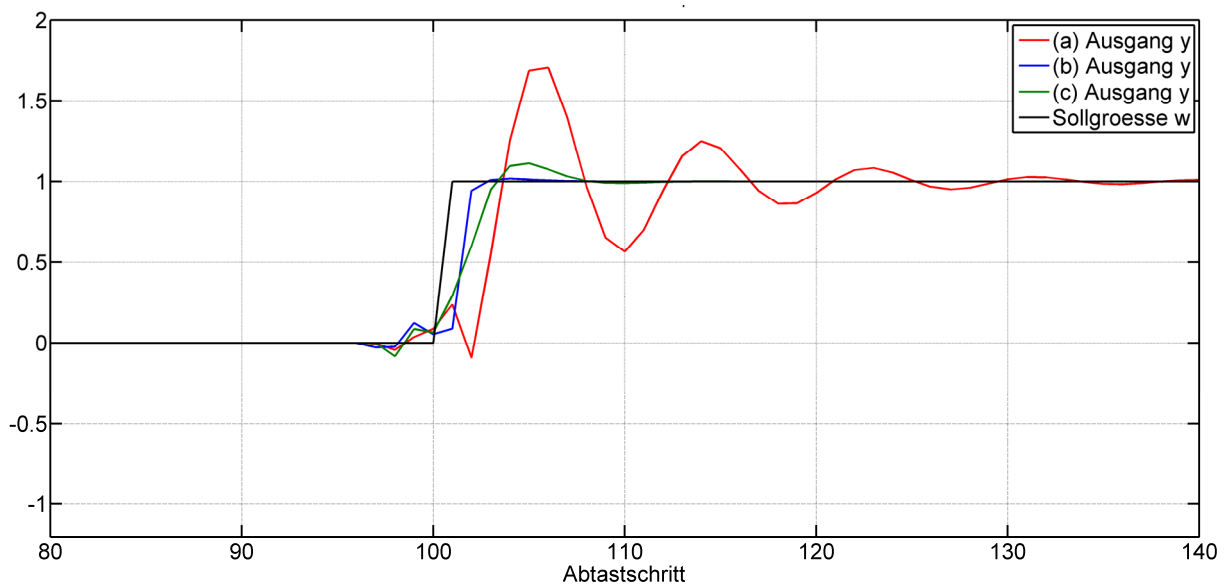


Abbildung 14: Antworten für einen Sprung in einem schwach-gedämpften System für drei verschiedene Parametersätze

In den vorangegangenen beiden Abbildungen sind exemplarisch Stellgröße und Systemausgang für ein schwach-gedämpftes System für drei Parametersätze (a), (b) und (c) dargestellt. Die Eigenfrequenz des gezeigten PT2-Systems war $f_0=10,0\text{Hz}$. Das LEHRsche Dämpfungsmaß D betrug $D=10,0 \cdot 10^{-3}$. Das System war nicht-minimalphasig mit Nullstellen bei 2,618 und 0,382. Die Parameter sind in Tabelle 4 angegeben.

	q_n	r_n	Kommentar
(a)	1	10	konservativer Einsatz der Stellgrößen, großer Fehler
(b)	10	1	früher Ausschlag der Steuergrößen; Systemantwort nahe am Sollgrößenverlauf
(c)	10	10	glatter Stellgrößenverlauf, dennoch akzeptables Ergebnis
Außerdem: $N_p=N_c=5$, $\tau=12\text{ms}$; keine Änderung in k			

Tabelle 4: Besetzung der Strafmatrizen für drei exemplarische Fälle

3.2.6.2 Prädiktionshorizont N_p

Die Wahl des Prädiktionshorizontes hängt von der Dynamik des Systems ab. Es lassen sich Richtwerte oder „Daumenregeln“ für die Wahl des Prädiktionshorizontes angeben, jedoch existieren keine geschlossenen algebraischen Richtlinien. Generell kann angegeben werden, dass eine Erhöhung des Prädiktionshorizontes eine stabilisierende Wirkung auf die Vorsteuerung hat (s. z.B. [23]).

➔ **Eine Erhöhung von N_p führt tendenziell zu einer Stabilisierung**

Wie jedoch aus den Abbildung 15 und Abbildung 16 hervorgeht, bedeutet eine Erhöhung des Horizontes nicht unbedingt eine Verbesserung des Regelverhaltens. Je größer der Horizont, desto früher sind Stellgrößenausschläge zu verzeichnen, auch wenn diese nicht nötig sind. Anzumerken ist, dass dieses Verhalten wie oben diskutiert auch von der Wahl der Strafparameter abhängt. Folgende Diagramme sind für das selbe schwach-gedämpfte System wie in Kapitel 3.2.6.1 aufgenommen worden. Für eine Wahl von $N_p < 2$ wurde die Vorsteuerung instabil. Weitere Parameter sind der Bildunterschrift zu entnehmen.

Die beiden folgenden Schaubilder (Abbildung 15 und Abbildung 16) verdeutlichen, dass bei gleichbleibendem $N_c=1$ und Variation von N_p das anfangs stark ausschlagende Regelgesetz mit zunehmendem N_p „gebändigt“ wird. Jedoch wird auch deutlich, dass ab einem gewissen Punkt (hier $N_p > 3$) das Steuerungsgesetz zu keiner Verbesserung des Ergebnisses, sondern eher zu einer Verschlechterung führt.

Für dasselbe System, jedoch mit $N_c=N_p$ und $R=0,10 \cdot Q$, ergibt sich ein anderes Bild, welches in Abbildung 17 und Abbildung 18 gezeigt ist. $N_p=2$ wird hier instabil, deshalb ist dieser Fall nicht mehr eingezeichnet.

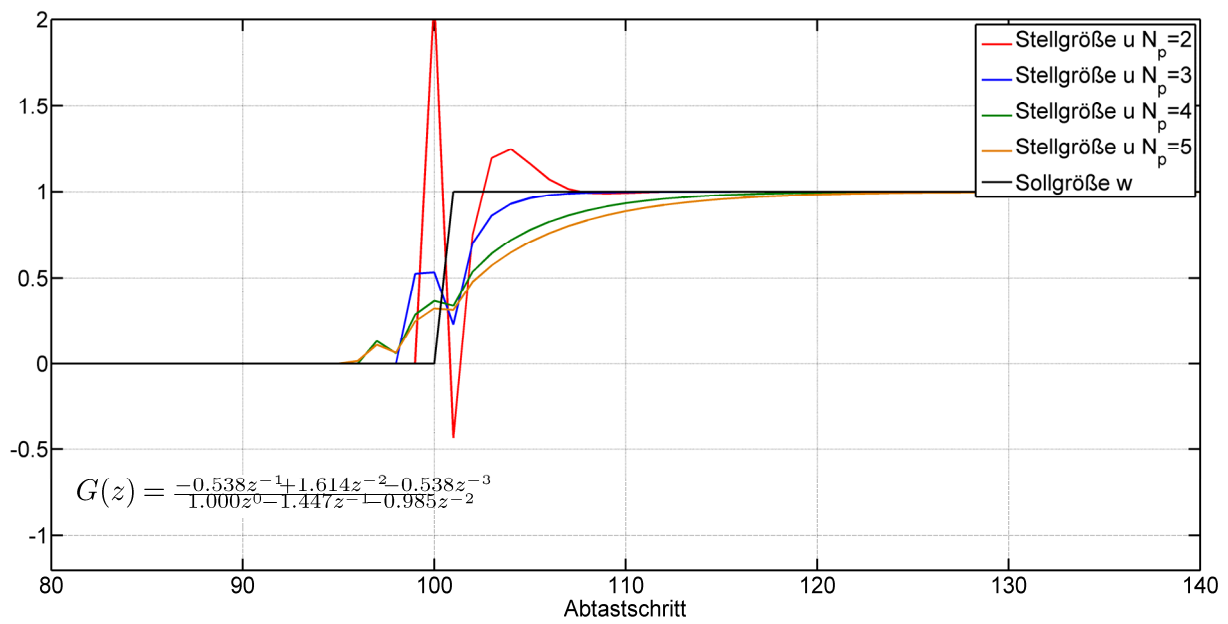


Abbildung 15: Stellgrößen für einen Sprung in einem schwach-gedämpften System für verschiedene Prädiktionshorizonte ($N_c=1$, $q^k=1,00$, $r^k=0,00$)

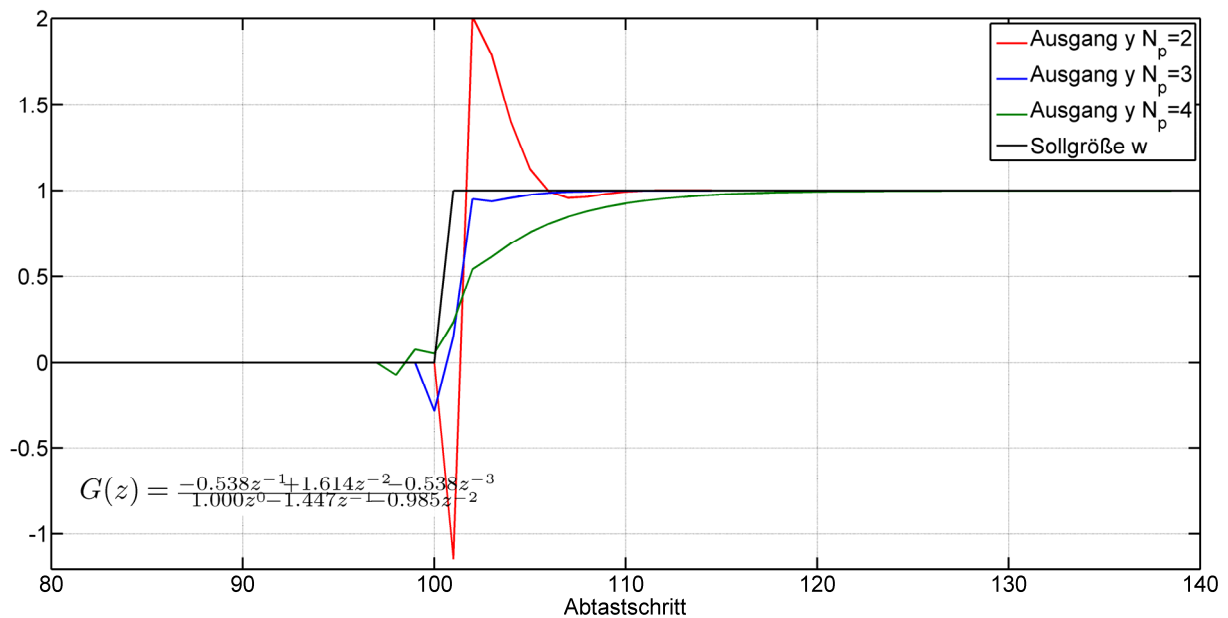


Abbildung 16: Systemausgang für einen Sprung in einem schwach-gedämpften System für verschiedene Prädiktionshorizonte

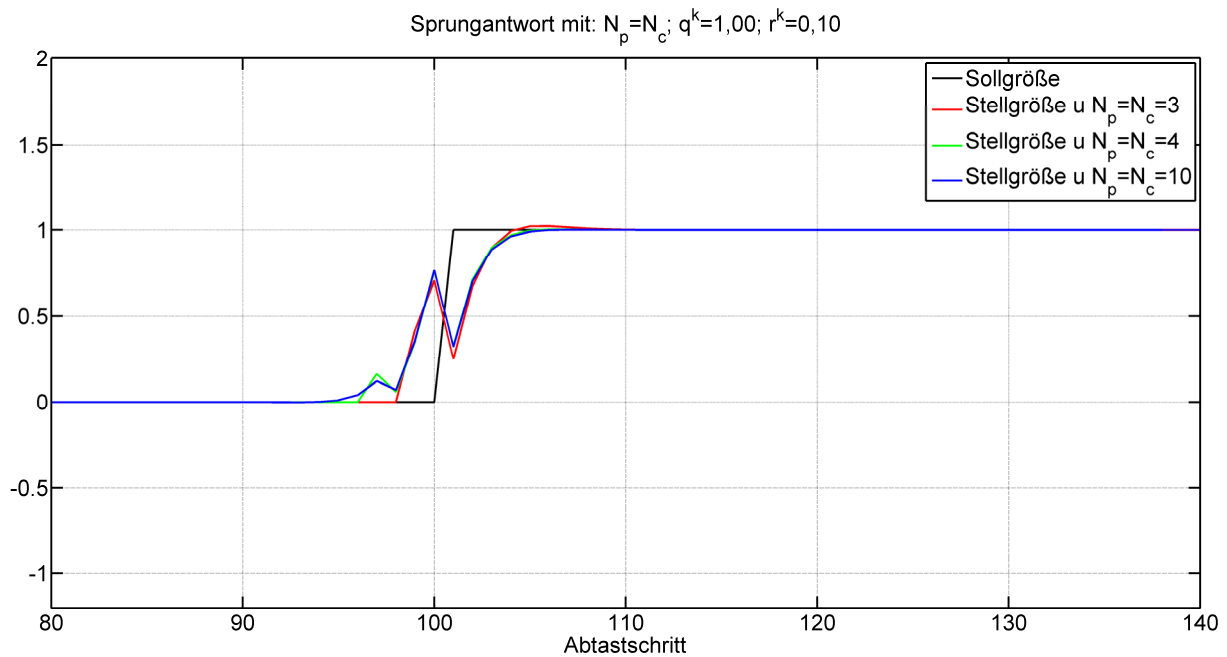


Abbildung 17: Stellgrößen für einen Sprung bei Variation von $N_p = N_c$

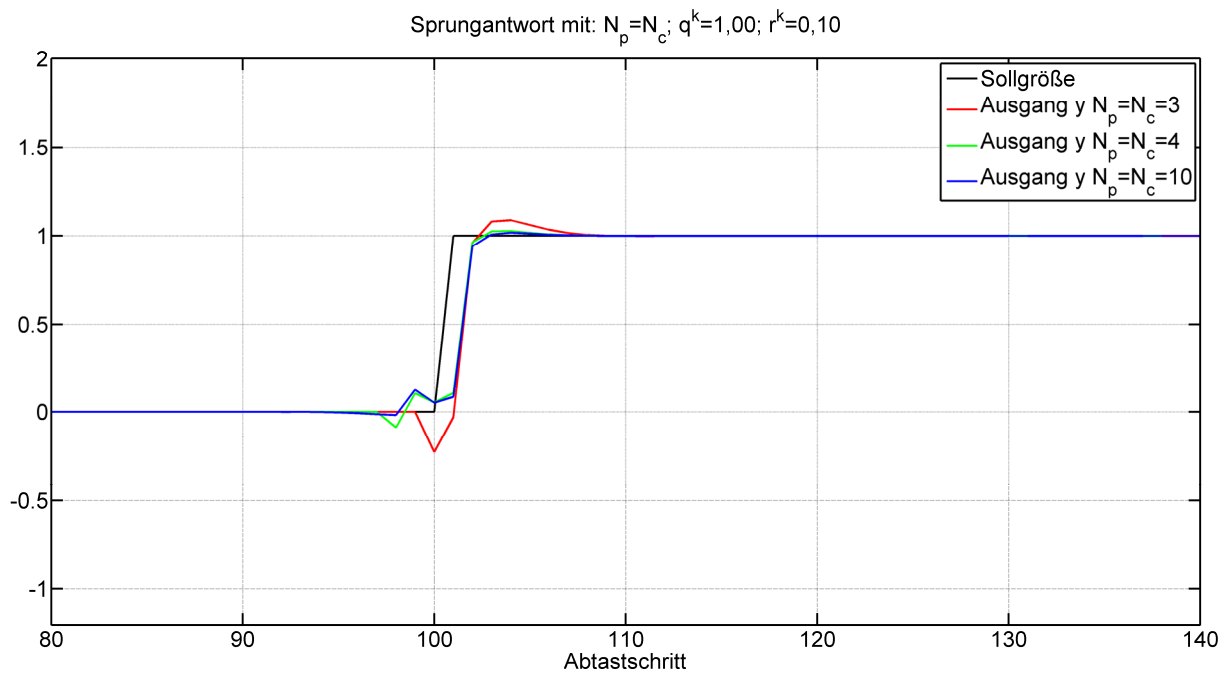


Abbildung 18: Systemausgang für einen Sprung bei Variation von $N_p = N_c$

Bestraft man, wie hier, die Steuerausschläge durch ein $R \neq 0$, so zeigt das Steuergesetz wie erwartet weniger Aktivität, das Ergebnis (Ausgang) scheint jedoch relativ gut auszufallen. Eine Erhöhung von N_p über 4 hinaus ergibt aber auch wieder

keine weitere Verbesserung des Ergebnisses. Allenfalls wird der erste Aktivitätszeitpunkt nach vorne verschoben, was selten von Vorteil ist.

Eine Richtlinie zur Wahl von N_p für den SISO-Fall geben CLARK et al. in [24]. Dabei sollte N_p mindestens dem Zählergrad (bei obigem Beispiel 3) entsprechen. Bei einem nicht-minimalphasigen System, bei dem die Sprungantwort anfangs negativ verläuft, sollte der Prädiktionshorizont mindestens den negativen und den nachfolgenden positiven Antwortverlauf „überblicken“. Jedoch geben sie an, dass $N_p \cdot T$ typischerweise der Anstiegszeit der Übertragungsfunktion entsprechen sollte. Diese Einschätzung deckt sich weitgehend mit den vorangegangenen Diagrammen. Für ein PT2-System lässt sich die Anstiegszeit t_a wie folgt berechnen:

$$t_a = \frac{1}{\omega_0} \cdot \frac{1}{\sqrt{1-D^2}} \cdot \left(\pi - \arctan\left(\frac{\sqrt{1-D^2}}{D}\right) \right) \quad (34)$$

Hierbei sind ω_0 die Eigenkreisfrequenz und D das LEHRsche Dämpfungsmaß.

Die Wahl des Horizontes kann aber auch noch von anderen Faktoren abhängen, die unabhängig von der prädiktiven Steuerung sind. Beispielsweise benötigt die interne Robotervorsteuerung des hier benutzen KUKA Roboters das Stellsignal \underline{u}_+ für ca. 22 Schritte in die Zukunft. Da stets $N_p \geq N_c$ gelten muss, wäre folglich $N_p \geq 22$.

3.2.6.3 Steuerungshorizont N_c

Hilfen zur Wahl von N_c sind ebenfalls in [24] gegeben. Eine Richtlinie besagt, dass N_c – insbesondere für hochdynamische Systeme – dem Systemgrad entsprechen sollte. Für ein System aus M Differenzengleichungen des Grades N ergibt sich somit

$$N_c \geq M \cdot N. \quad (35)$$

Eine weitere Leseart besagt, dass N_c mindestens der Anzahl der Systempole nahe der Stabilitätsgrenze entsprechen sollte. Für ein schwach-gedämpftes PT2-System entspricht dies im Allgemeinen der Richtlinie aus Gleichung (35). Tendenziell sollte also für hoch-dynamische Systeme ein größerer Steuerhorizont verwendet werden.

Ein niedriger Wert für N_c erwirkt ein eher träges Steuerverhalten, während eine Erhöhung zu aktiverem Stelleingriff führt. Dies ist damit zu erklären, dass die $N_p - N_c$ Einträge, die in \mathbf{R} fehlen, als ∞ betrachtet werden können [25].

➔ **Eine Verringerung von N_c führt tendenziell zu einer Stabilisierung**

4 Realisierung

4.1 Systemmodellierung

Im letzten Kapitel wurde auf die Herleitung des prädiktiven Regelgesetzes eingegangen. Ein Vorsteuerung kann per se jedoch nur erfolgreich sein, wenn das zugrunde liegende Modell der Strecke, die gesteuert/geregelt werden soll, ausreichend gut mit dem realen Verhalten des System übereinstimmt. In diesem Kapitel wird deshalb auf die Modellierung der Strecke eingegangen. Ergebnis wird ein mathematisches Modell $H(z)$ sein, welches zur Herleitung des Steuergesetzes nach Gleichung (5) verwendet werden kann.



Abbildung 19: Fotografie des Endeffektors K2

4.1.1 Grundlagen

Grundsätzlich müssen vor Beginn der Abstraktion und der Identifikation einige Überlegungen angestellt werden. Zwar soll, wie schon eingangs erwähnt, das Verhalten der Werkzeugspitze (TCP) gesteuert werden. Dazu wäre die Lage des TCP $\underline{p}_{tcp}^{ist}$ als Ausgang des zu identifizierenden Systems zu definieren. Dies hat sich jedoch als nachteilig erwiesen, da am TCP eine Überlagerung verschiedener Freiheitsgrade auftritt. Dies erschwert die Identifikation erheblich. Als „Ersatzausgang“ wurde deshalb ein Punkt gewählt, aus dem direkt auf die Lage des TCP geschlossen werden kann, an dem die Kopplungen aber minimal sind. Dies trifft sehr gut für den Mittelpunkt der Federn (*spring center point*, SCP) zu. Es ist letztlich unerheblich, ob der Roboter die auf den SCP oder die auf den TCP bezogene Trajektorie abfährt – wenn beide Punkte in einem bekannten Verhältnis stehen (in Abbildung 21 die farblichen Abstände), folgt aus „gutem Abfahren der SCP-Trajektorie“ direkt das „gute Abfahren der TCP-Trajektorie“. Der Identifikationsaufwand wird jedoch erheblich gesenkt. Auf Details dazu wird in Kapitel 4.1.3 genau eingegangen, eine Verdeutlichung des Vorgehens ist aber in folgender Abbildung zu sehen:

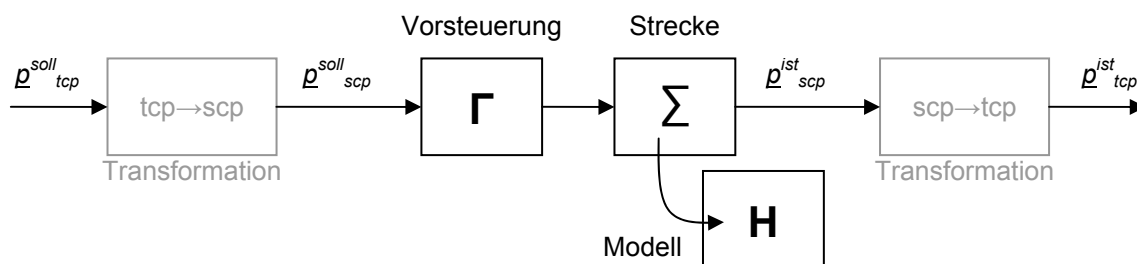


Abbildung 20: Umdefinition des Systemausgangs zum SCP

Ungeklärt ist noch, welches Signal von der Vorsteuerung erzeugt und dann in die Strecke eingespeist wird. Als Systemeingang des Roboters lässt sich im Prinzip jeder körperfeste Punkt des letzten Robotergliedes definieren. Die kinematische Transformation der Robotersteuerung wandelt kartesische Koordinaten dieses Punktes über die bekannten Abmaße in Achskoordinaten um (vgl. auch Kapitel 3.1.1). Da das Messsignal des Kraft-Momenten-Sensors, der einseitig fest mit dem letzten Roboterglied verbunden ist, gerade die relative Lage zwischen unausgelenktem (SCP0, „Nominallage“) und ausgelenktem (SCP) Federmittelpunkt darstellt, eignet sich der SCP0 als Robotereingang sehr gut. Dieser Sachverhalt ist in Abbildung 21 dargestellt.

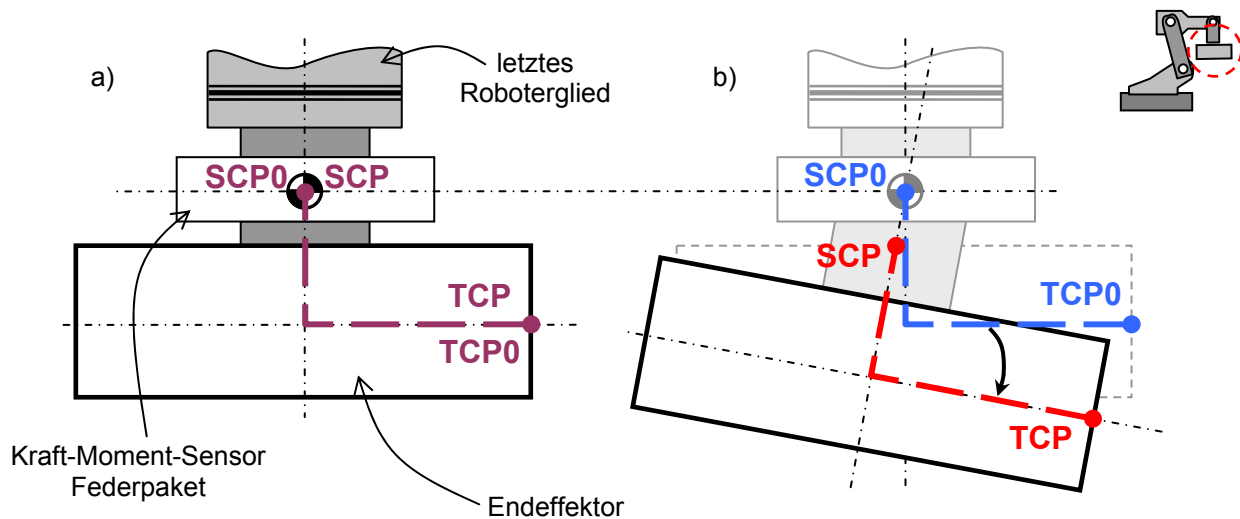


Abbildung 21: Skizze zur Verdeutlichung der Lagen von TCP und SCP im
(a) unausgelenkten Nominalfall und (b) ausgelenkten Fall

Der SCP0 wird somit als körperfest mit dem letzten Roboterglied verbunden angesehen. Der SCP ist der Mittelpunkt der ausgelenkten Federn und ist fest mit dem Endeffektor verbunden. Somit stellt $\mathbf{R}(z)$ als Roboterübertragungsfunktion einen Zusammenhang zwischen Soll- und Istlage des SCP0 ($\underline{p}_{0,scp0}$) her. Die Komplettierung der kinematischen Kette bis zum Systemausgang $\underline{p}_{0,scp}$ wird durch die Endeffektordynamik $\mathbf{E}(z)$ beschrieben. Somit wird:

$$\underline{p}_{0,scp}^{ist} = \mathbf{E}(z) * \underline{p}_{0,scp}^{ist} = \mathbf{E}(z) \cdot \left(\mathbf{R}(z) * \underline{p}_{0,scp0}^{soll} \right) = \underbrace{\left(\mathbf{E}(z) \cdot \mathbf{R}(z) \right)}_{=\mathbf{H}(z)} * \underline{p}_{0,scp0}^{soll} \quad (36)$$

Die folgenden Kapitel befassen sich also mit der Identifikation der beiden Übertragungsfunktionen $\mathbf{R}(z)$ und $\mathbf{E}(z)$.

4.1.2 Roboterverhalten $\mathbf{R}(z)$

Ein 6-Freiheitsgrade Roboter ist ein komplexes System aus Antrieben, Gelenken, Getrieben und elektrischen Komponenten mit jeweils eigenem dynamischen Verhalten. Ein Großteil dieses Verhaltens wird durch interne Vorsteuerungen und Regelungen kompensiert. Dennoch kann ein Roboter nicht als „ideales“ System angesehen werden, welches alle kommandierten Größen auch ohne Verzögerungen

ausführt. Den Roboterblock aus Abbildung 8 kann man zur Verdeutlichung noch weiter aufteilen:

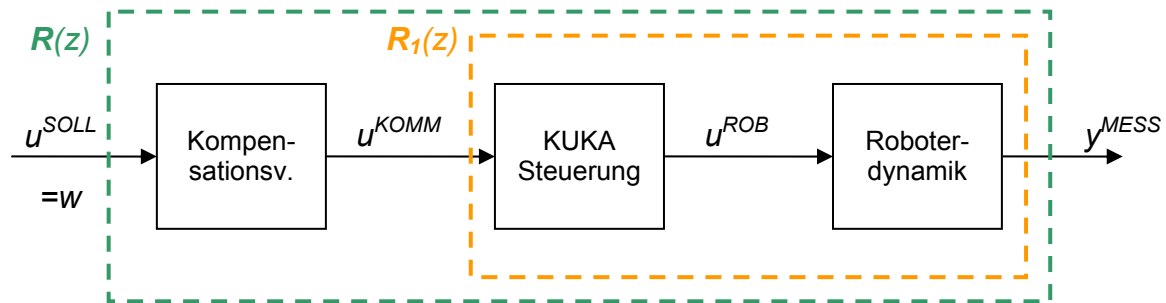


Abbildung 22: Detailansicht des dynamischen Systems „Roboter“

Die *Kompensationvorsteuerung* wurde in einer vorhergehenden Arbeit entwickelt, um Auswirkungen des Endeffektors auf die Genauigkeit des Roboters und das Totzeitverhalten zu verbessern. Sie basiert auf einer iterativ-trainierten, achsbasierten adaptiven Vorsteuerung auf FIR-Filterbasis [19]. Durch die Achsbasierung und die vergleichsweise hohe Anzahl an Koeffizienten bei FIR-Filtern war die Übertragungsfunktion nicht bekannt.

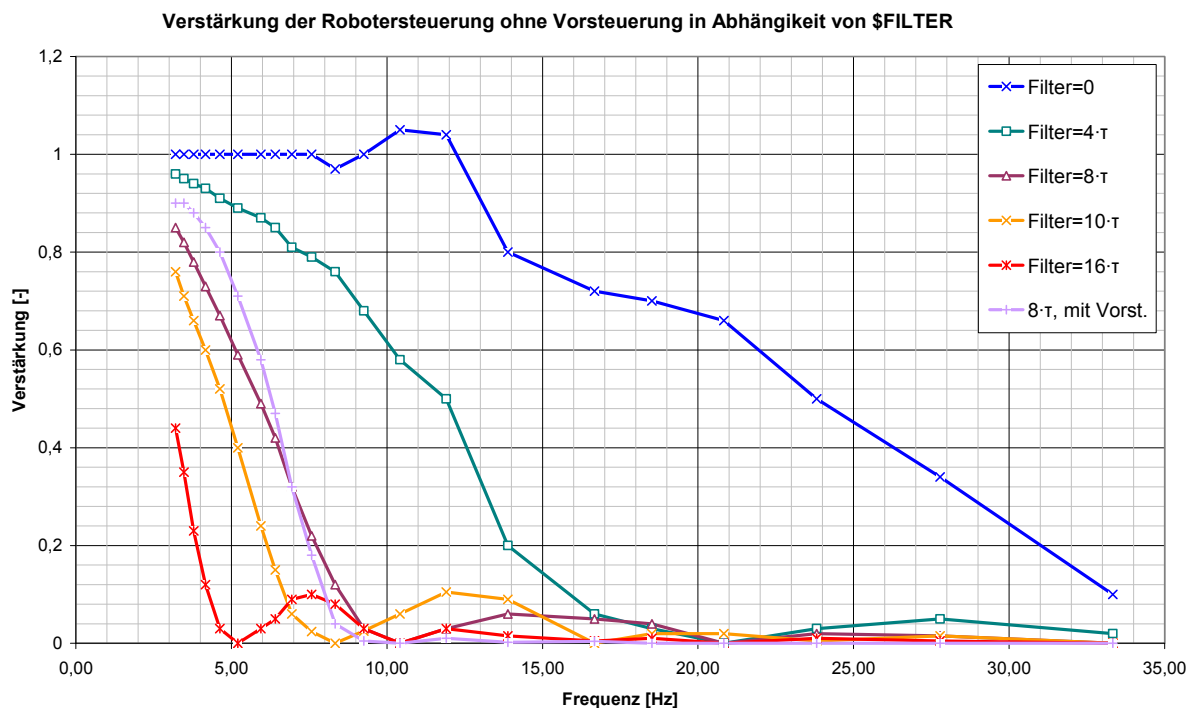


Abbildung 23: Gemessene Übertragungsfunktion von A6 des Roboters
in Abhängigkeit von \$FILTER für $f \geq 3,00$ Hz

Die KUKA-interne Vorsteuerung ist in erster Linie ein Mittelwertfilter der Fensterbreite F . Die Fensterbreite ist in der KUKA-Steuerung einstellbar und weist ein Übertragungsverhalten auf, wie es in Abbildung 23 darstellt ist, auf. Man sieht gut, dass für die meisten Filterbreiten der Durchlassbereich oberhalb ca. 7,0 Hz um mehr als 50% gedämpft ist. Nur bei Deaktivierung oder einer Fensterbreite von 48ms ($4 \cdot \tau$) ist Durchlass in einem größeren Bereich möglich. Jedoch sollte beachtet werden, dass das Mittelwertfilter dazu dient, zu hohe Belastungen von den Motoren und Getrieben zu nehmen. Zu hohe Frequenzen im Eingang bewirkten große Stellmomente, was auf Dauer die Lebenszeit verkürzt.

Ein zu hohes Eingangsfiler ist gut für die Lebensdauer des Roboters, jedoch können damit keine Stellgrößen über ca. 7,0 Hz vom Roboter ausgeführt werden. Ein niedriges Filter lässt den Roboter auch schnelle Bewegungen durchführen, jedoch auf Kosten der Bauteillebensdauer. Die diskrete Übertragungsfunktion eines Mittelwertfilters der Breite F lautet:

$$M_F(z) = \frac{1}{F} \cdot (z^0 + z^{-1} + \dots + z^{-F}) \quad (37)$$

Es ist wichtig anzumerken, dass $M_F(z)$ genau F Nullstellen auf dem Einheitskreis hat und somit nicht-minimalphasig ist. Eine Inversion führt zu starkem Schwingverhalten. Dies muss bei der Parameterwahl nach Kapitel 3.2.6 berücksichtigt werden. Der Einfluss wurde in den Experimenten berücksichtigt.

Das letzte Glied ist der eigentliche Roboter, in dem alle Trägheiten, Motoren, Getriebe und elektronische Schaltkreise zusammengefasst sind. Messungen haben gezeigt, dass der Amplitudeneinfluss des Mittelwertfilters meist über die Roboterdynamik überwiegt (vor allem bei **K2**). Von Amplitudenseite her konnte der Roboter so weitgehend als ideal betrachtet werden. Somit kann der gesamte (in Abbildung 22 orangene) Block **R₁** als Mittelwertfilter angenähert werden. Jedoch muss berücksichtigt werden, dass der Roboter eine Totzeit von ca. 3 Zeitschritten umfasst. Abhängig von der Filterbreite F ergibt sich die Roboterübertragungsfunktion näherungsweise zu:

$$\mathbf{R}(z) \approx \mathbf{1}_{M \times M} \cdot M_F(z) \cdot z^{-3} \quad (38)$$

4.1.3 Endeffektorverhalten $E(z)$

Die Endeffektorübertragungsfunktion stellt einen Zusammenhang zwischen der Lage des SCP0 (als Ausgang von $R(z)$) und der Lage des SCP her. Wie in Abbildung 21 gezeigt ist, wird angenommen, dass sich der Endeffektor rotatorisch um den SCP0 auslenken kann. Die *relative* Beziehung zwischen TCP und SCP bzw. TCP0 und SCP0 bleibt jedoch immer erhalten. Somit ist der SCP definiert als derjenige Punkt, für den gilt:

$${}^{\#}\mathbf{T}_{scp} = {}^{\#}\mathbf{T}_{tcp} \cdot \left(\underbrace{{}^{scp0}\mathbf{T}_{tcp0}}_{\text{konstant}} \right)^{-1} \quad (39)$$

Die Lage des TCP hängt also über die in ${}^{scp0}\mathbf{T}_{tcp0}$ enthaltenen festen Hebelarme von der Lage am SCP0 ab. Insbesondere die Positionen des TCP sind über die Hebelarme mit den Relativwinkeln überlagert und kleine Orientierungsänderungen können zu größeren Positionsabweichungen führen. Diese nichtlinearen (i. d. R. trigonometrischen) Kopplungen erschweren insbesondere die dynamische Systemidentifikation und verkomplizieren die Implementierung. Daher wird wie oben beschrieben jede auf den TCP bezogene Größe nach (39) zum SCP umgerechnet, an dem diese Kopplungen weit weniger ins Gewicht fallen. Damit lautet die Kette der homogenen Transformationsmatrizen vom Ursprung bis zum tatsächlichen SCP:

$${}^W\mathbf{T}_{scp} = {}^W\mathbf{T}_{scp0} \cdot {}^{scp0}\mathbf{T}_{scp} \quad (40)$$

Der erste Term ${}^W\mathbf{T}_{scp0}$ lässt sich mit Hilfe der Roboterkinematik berechnen und ist somit bekannt. Der Kraft-Momenten-Sensor selbst liefert direkt ${}^{scp0}\mathbf{T}_{scp}$. Somit die Lage des SCP direkt messbar. Der Übergang eines für den TCP vorliegenden Signals zum SCP funktioniert wie in (39) gezeigt. Dadurch, dass nahe der Nulllage zwischen dem Weltkoordinatensystem W und dem $scp0$ -System ungefähr $\beta=90^\circ$ herrscht, müssen zur Umrechnung der homogenen Transformationsmatrizen in Lagevektoren diese zunächst in ein Koordinatensystem transformiert werden, in dem die Winkeldarstellung nicht singulär ist (vgl. Abbildung 7). Geeignet dazu ist das R System, das – wie in Abbildung 6 gezeigt – auch im Ursprung 0 liegt, jedoch 90° gegenüber dem W -System verdreht ist.

Zur Erstellung eines mathematischen Modells des Endeffektorverhaltens ist Abstraktion nötig. Transformiert man Gleichung (40) ins \mathbf{R} -System und anschließend in vektorielle Darstellung so ergibt sich:

$${}^R \underline{p}_{0,scp} = \underbrace{{}^R \underline{p}_{0,scp0}}_{\text{roboterseitig}} + \underbrace{{}^R \underline{p}_{scp0,scp}}_{\text{endeffektorseitig}} \quad (41)$$

Im Weiteren wird auf die Angabe des Koordinatensystems verzichtet. Falls nicht anders angegeben, liegt ein Lagevektor im \mathbf{R} -System vor. Außerdem gelte zur Bewahrung der Übersichtlichkeit folgende Abkürzung (\underline{s} wie \underline{s} ensorausgang):

$$\underline{p}_{scp0,scp} := \underline{s} \quad (42)$$

Anfangs soll der Term \underline{s} genauer untersucht werden. Ändert sich die Lage des Roboters (bspw. durch ein Anstellen des Nickwinkels), so wird sich die Lage des Endeffektors relativ zum Roboter abhängig von der Gewichtskraft neu einstellen. Zusätzlich werden durch die Beschleunigungen Schwingungen induziert. Es gilt deshalb folgende Annahme: Die Relativlage \underline{s} des Endeffektors gegenüber dem Roboter setzt sich zusammen aus:

- ... einem statischen Anteil aufgrund der Gewichtskraft und der Orientierung des letzten Robotergliedes (repräsentiert durch \underline{p}_{scp0})
- und einem dynamischen Anteil aufgrund der Beschleunigungen am SCP0

Dies ist in folgender Gleichung zusammengefasst:

$$s(t, \underline{p}_{scp0}) = \underbrace{f(\underline{p}_{scp0})}_{\text{statisch}} + \underbrace{\underline{s}^{dyn}(t)}_{\substack{\text{dynamisch} \\ 0 \Leftrightarrow t \rightarrow \infty}} \quad (43)$$

Der dynamische Anteil wird in Abschnitt 4.1.3.2 behandelt. Da die Steifigkeiten, Anschlagpositionen und Nichtlinearitäten des Kraftmomentensensors nicht hinreichend bekannt sind, kann der statische Anteil nicht analytisch abgeleitet werden. Als Ansatz wurde deshalb eine Annäherung dieser Funktion durch eine TAYLOR-Entwicklung in einer stationären Nulllage \underline{p}_{scp0}^0 gewählt.

$$f(\underline{p}_{scp0}) = f(\underline{p}_{scp0}^0) + \nabla f(\underline{p}_{scp0}^0) \cdot (\underline{p}_{scp0} - \underline{p}_{scp0}^0) + THO \quad (44)$$

Hierbei seien THO die Terme höherer Ordnung. OBdA wird die Nulllage \underline{p}_{scp0}^0 zum Nullvektor $\underline{0}$ gewählt. Nach Abbruch der Reihe nach dem linearen Anteil vereinfacht sich (43) wie folgt:

$$\begin{aligned}\underline{s}(t, \underline{p}_{scp0}) &\approx \underbrace{f(\underline{p}_{scp0}^0)}_{\underline{s}^0} + \underbrace{\nabla f(\underline{p}_{scp0}^0)}_{\mathbf{J}_0} \cdot \underline{p}_{scp0} + \underline{s}^{dyn}(t) \\ &= \underline{s}^0 + \mathbf{J}_0 \cdot \underline{p}_{scp0} + \underline{s}^{dyn}(t)\end{aligned}\quad (45)$$

Hierbei ist \mathbf{J}_0 die JACOBI-Matrix in der Ruhelage. Der erste Term \underline{s}^0 ist dabei gerade die Auslenkung des Endeffektors durch das Gewicht im Ruhezustand, während der zweite Term in erster Linie die Lageänderung des Endeffektors in Abhängigkeit der Orientierungsänderung des SCP0s beschreibt.

Allgemein kann nun geschrieben werden:

$$\underline{s}(t, \underline{p}_{scp0}) = \underline{s}^{stat}(\underline{p}_{scp0}) + \underline{s}^{dyn}(t) \quad (46)$$

4.1.3.1 Statische Systemidentifikation

Im Folgenden wird auf die Identifikation der statischen Terme in (45) eingegangen.

Statischer Anteil 0. Ordnung

Der Term \underline{s}^0 wird auch als statischer Anteil 0. Ordnung bezeichnet. Er ist lediglich die Differenz zwischen der Nominallage des Endeffektors und der tatsächlichen Lage im eingeschwungenen Zustand und kann damit direkt gemessen werden. Dazu muss nur das letzte Roboterglied in eine senkrechte Lage gefahren und das Signal des KMS ausgelesen werden.

Für die Roboterkonfiguration **K1** ergibt sich exemplarisch folgender Vektor. Der Endeffektor hängt also um ca. einen halben Millimeter nach unten und nickt leicht.

$${}_R \underline{s}^0 \approx \left(\begin{array}{c|c} \overbrace{\begin{matrix} 0,556 & 0,0 & 0,0 \end{matrix}}^{[mm]} & \overbrace{\begin{matrix} 0,0 & +0,430 & 0,0 \end{matrix}}^{[mrad]} \end{array} \right)^T$$

Statischer Anteil 1. Ordnung

Die Gewichtsauslenkung aufgrund einer nicht-waagerechten Lage des SCP0 wird hier mit einem *statischen Anteil 1. Ordnung* oder *Linearteil* bezeichnet. Im Detail geht es um die Besetzung der Matrix \mathbf{J}_0 mit Inhalt. Die Matrix hat folgende Struktur:

$$\mathbf{J}_0 = \begin{bmatrix} J^{x \rightarrow x} & J^{y \rightarrow x} & \dots & J^{\gamma \rightarrow x} \\ J^{x \rightarrow y} & J^{y \rightarrow y} & \dots & J^{\gamma \rightarrow y} \\ \vdots & \vdots & \ddots & \vdots \\ J^{x \rightarrow \gamma} & \dots & \dots & J^{\gamma \rightarrow \gamma} \end{bmatrix} \stackrel{\text{kurz}}{=} \begin{bmatrix} \ddots & & \\ & J^{mn} & \\ \ddots & & \ddots \end{bmatrix} \in \mathcal{R}^{M \times M} \quad (47)$$

Jeder Eintrag kann als linearer Skalierungsfaktor verstanden werden, wobei viele Einträge 0 sind. Ist der SCP0 um nur um $\beta_{0,scp0}$ aus der Ruhelage ausgelenkt, so „hängt“ der Endeffektor um

$$\beta_{scp0,scp} = J^{\beta\beta} \cdot \beta_{0,scp0} \quad (48)$$

„hinterher“. Dies ist in Abbildung 24 verdeutlicht.

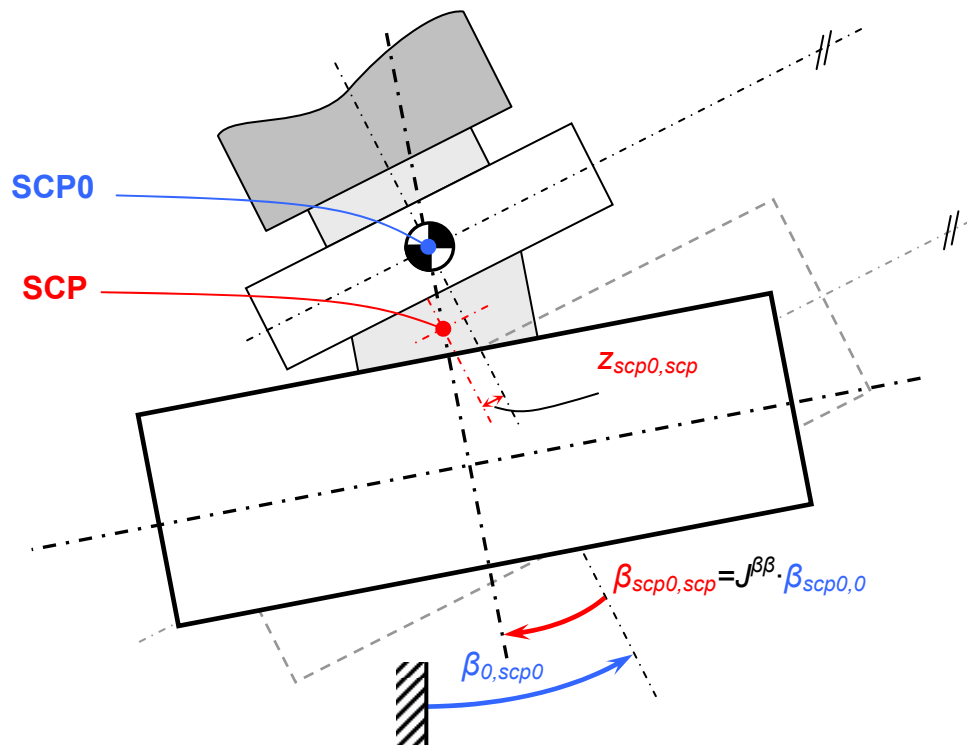


Abbildung 24: Darstellung der Auslenkung des Endeffektors bei nicht-waagerechter Lage des SCP0

Eine reine *Positions*änderung des Roboters kann aber keine Änderung der (relativen) Lage \underline{s} zum Ergebnis haben kann. Somit sind die ersten 3 Spalten von \mathbf{J}_0 jeweils 0. Auch wirken sich die Winkeländerungen am Roboter kaum auf die *Postions*abweichungen des SCP gegenüber des SCP0 aus. Beispielhaft ist dies in Abbildung 24 für $\mathbf{z}_{scp0,scp}$ gezeigt (tatsächlich liegt der Effekt im Zehntel-Millimeterbereich). Somit vereinfachen sich auch die ersten drei Zeilen von \mathbf{J}_0 zu 0.

Ist der Endeffektor in seiner Ausgangslage waagerecht (wie z.B. in **K1**), so hat auch eine Drehung γ um $_{scp0}x$ keinen Effekt auf irgendeinen anderen Freiheitsgrad, da die Drehachse parallel zur Erdbeschleunigung steht. Somit sind für **K1** auch Spalte und Zeile 6 jeweils 0. Übrig bleiben folgende Einträge:

$$\mathbf{J}_0 = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ J^{\alpha\alpha} & J^{\alpha\beta} & 0 \\ \mathbf{0}_{3 \times 3} & J^{\beta\alpha} & J^{\beta\beta} & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (49)$$

Es konnte experimentell gezeigt werden, dass auch die Einflüsse $J^{\alpha\beta}$ und $J^{\beta\alpha}$ vernachlässigbar gering sind. Durch die sukzessive Variation der verbliebenen Freiheitsgrade des Roboters und der jeweiligen Messung von \underline{s} durch den Kraft-Momenten-Sensor lassen sich die verbliebenen Einträge identifizieren. Für die Roboterkonfiguration **K1** ergibt sich exemplarisch für den Winkel α folgende Anpassung:

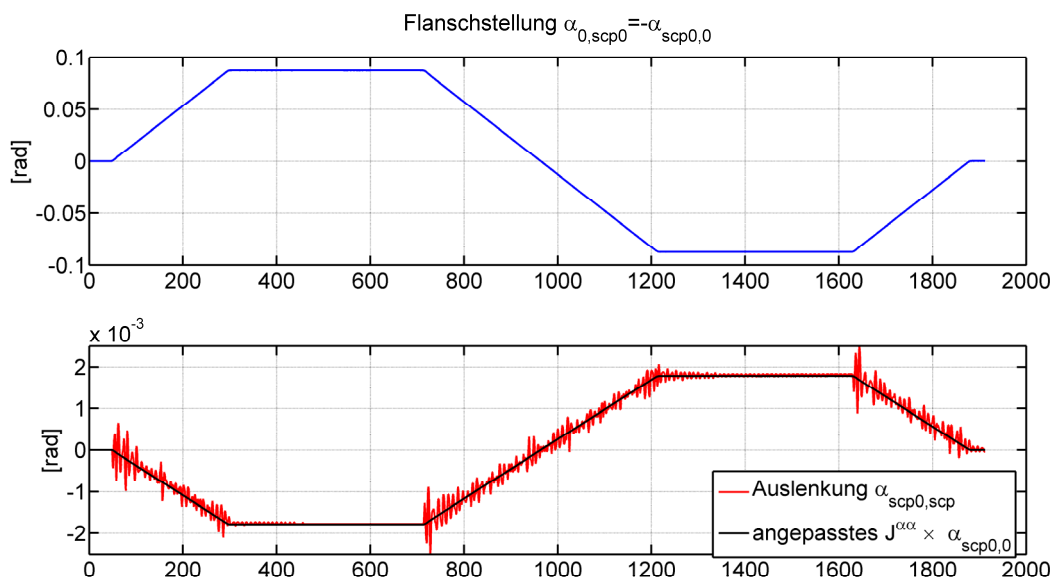


Abbildung 25: Anpassungsergebnis der Jacobi-Matrix für den Winkel α mit dem Faktor $J^{\alpha\alpha} = -20,2 \cdot 10^{-3}$

Die Annahme eines linearen Zusammenhangs ist zumindest für **K1** in diesem Winkelbereich ($\pm 0,10 \text{ rad} = \pm 5,73^\circ$) durchaus gerechtfertigt. Anzumerken ist auch noch, dass $J^{\alpha\alpha} = -20,2 \cdot 10^{-3}$ fast identisch mit $J^{\beta\beta} = -20,4 \cdot 10^{-3}$ ist, was wohl auf vergleichbare Steifigkeiten des Sensors in diesen Freiheitsgraden zurückzuführen ist. Die maximale Auslenkung von $y_{scp0,scp}$ aufgrund von $\alpha_{0,scp0}$ war mit $\pm 0,16 \text{ mm}$ in der Tat zu vernachlässigen. Für andere Freiheitsgrade fiel die Auslenkung noch geringer aus.

4.1.3.2 Dynamische Systemidentifikation

Immer noch unbekannt ist das dynamische Verhalten $\underline{s}^{dyn}(t)$ des Endeffektors. Hier kann nicht mit einer einfachen statischen Identifikation vorgegangen werden. Vielmehr muss eine dynamische Übertragungsfunktion $\mathbf{E}_{dyn}(z)$ gefunden werden, die einen Zusammenhang zwischen äußerer Anregung und Schwingungsverhalten des Endeffektors herstellt. Wie schon in Kapitel 4.1.1 festgestellt, wird als Eingang von $\mathbf{E}_{dyn}(z)$ die Lage des SCP0 definiert. Dieser Zusammenhang ist in Gleichung (50) verdeutlicht.

$$\underline{s}^{dyn}(t) = \mathbf{E}_{dyn}(z) * \underline{p}_{scp0}(t) \quad (50)$$

Es gibt verschiedene Möglichkeiten, eine dynamische Übertragungsfunktion zu identifizieren. Eine weit verbreitete Möglichkeit ist die Vorgabe eines „leeren“ parametrischen Systemmodells (sog. „gray box model“). Dessen Parameter werden durch geeignete Optimierungsverfahren derart angepasst, dass der Ausgang des Modells möglichst deckungsgleich mit dem gemessenen Ausgangssignal ist. Wichtig dabei ist, dass das (zu identifizierende) System mit Signalen angeregt wird, die möglichst das gesamte interessante Frequenzspektrum der Anwendung umfassen. Der Ingenieur hat die Aufgabe, sowohl die Grobstruktur des Modells, als auch die Anregungssignale möglichst *systemnah* zu wählen. Zusätzlich muss ein Optimierungsverfahren ausgewählt werden.

Im Folgenden soll kurz auf die Modellstruktur, die Anregungssignale und ein Optimierungsverfahren eingegangen werden. Die genaue Erläuterung der Identifikation sprengt jedoch den Rahmen dieser Abfassung.

Modellstruktur

Das hier betrachtete System hat $M=6$ Freiheitsgrade. Ähnlich wie \mathbf{J}_0 bildet $\mathbf{E}_{dyn}(z)$ das Verhalten einer 6-dimensionalen Lage (Eingang \underline{p}_{scp0}) auf eine andere 6-dimensionale Lage (Ausgang \underline{s}) ab. $\mathbf{E}_{dyn}(z)$ besteht also aus $M^2=36$

Einzelübertragungsfunktionen, die jeden Eingangsfreiheitsgrad auf alle Ausgänge überträgt. Manche werden 0 sein – dort spricht man von nicht-vorhandener Kopplung. Im Gegensatz zu \mathbf{J}_0 hingegen sind die einzelnen Elemente keine skalaren Größen, sondern Übertragungsfunktionen (ÜFen) des diskreten Zeitverschiebeoperators z . Jede ÜF von einem Eingang n zu einem Ausgang m besteht aus einem Zähler- und einem Nennerpolynom mit folgender Form:

$$E_{mn}(z) = \frac{b_{mn}^0 \cdot z^{-0} + b_{mn}^1 \cdot z^{-1} + \dots + b_{mn}^{N_b} \cdot z^{-N_b}}{a_{mn}^0 \cdot z^{-0} + a_{mn}^1 \cdot z^{-1} + \dots + a_{mn}^{N_a} \cdot z^{-N_a}} \cdot z^{D_{mn}} = \frac{y_m}{u_n} \quad (51)$$

Zähler- und Nennergrad sind entsprechend der Notation N_b bzw. N_a . Zusätzlich kann eine Totzeit vorhanden sein, die die Wirkung des Eingangs um D_{mn} Zeitschritte verzögert. Das Zählerpolynom enthält dabei die Informationen, wie der Eingang u_n auf den Ausgang wirkt (bspw. ob das Eingangssignal differenziert wird). Im Nenner sind die Informationen enthalten, wie der Systemausgang y_n auf Eingangssignale reagiert. Insbesondere sind bei einem einfachen schwingungsfähigen System in den a -Parametern die Eigenfrequenz und –dämpfung enthalten. Jeder Freiheitsgrad des Endeffektors ist ein Systemausgang. Es wird hier angenommen, dass sich das Schwingverhalten durch einfache PT2-Glieder ($N_a=2$) annähern lässt. Damit hängen die Eigenschaften der jeweiligen Schwingungen nur von der Masse, den Federsteifigkeiten der Federn und den Dämpfungseigenschaften ab. In welcher Frequenz bzw. mit welchem Abklingverhalten ein Freiheitsgrad schwingt, hängt also alleine von diesen Faktoren ab – es ist dafür unerheblich, *wie* (d.h., durch welchen Eingang) die Schwingung angeregt wurde. Nur ein stationärer Übertragungsfaktor bestimmt die Stärke, wie ein Ein- auf einen Ausgang wirkt (insbesondere ist dieser Faktor 0 für ungekoppelte Freiheitsgrade, z.B. $y \rightarrow \beta$). Da die Schwingungseigenschaften in den Nennerpolynomen der Übertragungsmatrix steht, hat damit jede Zeile (=Ausgang) von $\mathbf{E}_{dyn}(z)$ ein und dasselbe Zählerpolynom.

Ebenso wurde angesprochen, dass nur eine Beschleunigung und dadurch eine Kraft eine Schwingung hervorrufen kann. Wählt man als Eingang p_{scp0} , so muss dieses doppelt abgeleitet werden, um auf die Beschleunigung zu kommen. Eine doppelte Differenzierung kann im diskreten Fall durch folgendes Übertragungspolynom approximiert werden:

$$\ddot{X} \approx K \cdot (1 \cdot z^0 - 2 \cdot z^{-1} + 1 \cdot z^{-2}) \cdot X \quad (52)$$

Damit ist $N_b=2$ festgelegt. Einzig der Skalierungsfaktor K bleibt als variable Größe vorhanden.

Somit wurden für $\mathbf{E}_{dyn}(z)$ folgende strukturelle Systemannahmen getroffen:

- Es existieren genau 6 Nennerpolynome, d.h. eines pro Freiheitsgrad und Zeile
- Sämtliche Nennerpolynome haben Grad $N_a=2$ (\rightarrow PT2-Systeme)
- Sämtliche Zählerpolynome haben die Form von (52) ($N_b=2$)
- Es verbleibt eingangsseitig nur der Verstärkungsfaktor K_{mn} als Parameter

Somit reduziert sich der Identifikationsaufwand erheblich.

Anregungssignale

Damit insbesondere die a -Parameter korrekt von einem Optimierungsverfahren angepasst werden können, sollte in den Anregungssignalen ein möglichst großes Spektrum an Frequenzanteilen vorhanden sein. Jedoch sollte mindestens das Eigenschwingungsverhalten der Freiheitsgrade abgebildet werden können. Somit sollten die Signale auf jeden Fall ausreichend Leistungsanteile im Bereich der Systemeigenfrequenzen enthalten.

Die Eigenfrequenzen können vor Beginn der eigentlichen Identifikation abgeschätzt werden. Sind Massen, Trägheitsmomente und Steifigkeiten grob bekannt, können die Eigenfrequenzen errechnet werden (s. z. B. [26]). Auch kann das System aus seiner Ruhelage ausgelenkt und plötzlich losgelassen werden. Die Eigenfrequenzen (und Dämpfungen) können aus der abklingenden Schwingung gewonnen bzw. abgeschätzt werden. Für die Roboterkonfiguration **K1** lagen alle Eigenfrequenzen in einem Bereich zwischen 6,50–12,50 Hz. Somit sollte das Anregungssignal zumindest diesen Frequenzbereich umfassen.

Als Signale eignen sich beispielsweise stochastische Signale. Diese enthalten ein gleiches Leistungsspektrum über große Frequenzbereiche. Jedoch sind diese schwer durch einen Roboter auszuführen, zumal die Amplituden meist sehr klein sind. Besser geeignet sind hier so genannte Chirp-Signale (von engl. *to chirp* – *zwitschern*), die ein Sinussignal mit ansteigendem Frequenzanteil haben. Für ein Signal mit linear ansteigender Frequenz ist dies in folgender Gleichung gegeben:

$$x_{chirp}^{lin}(t) = A \cdot \sin \left(2\pi \cdot \underbrace{\left(f_0 + \frac{k}{2}t \right)}_{f(t)} \cdot t \right) = A \cdot \sin(\omega(t) \cdot t) \quad (53)$$

Hierbei ist A die Schwingungsamplitude, f_0 die Startfrequenz und k die Steigerungskonstante. Zur Wahl von k in Abhängigkeit der maximalen Frequenz und dem Endzeitpunkt besteht folgender Zusammenhang:

$$k = 2 \cdot \frac{(f_{\max} - f_0)}{t_{\text{end}}} \quad (54)$$

Exemplarisch ist ein Chripsignal in Abbildung 26 gezeigt:

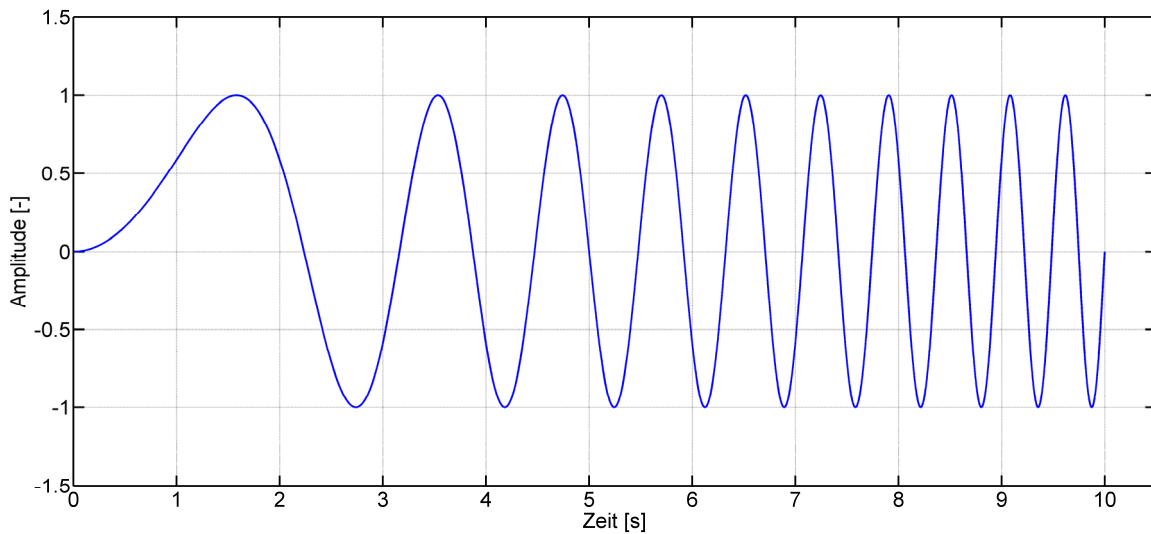


Abbildung 26: Lineares Chripsignal mit $f_0=0,00 \text{ Hz}$ und $k=0,20 \text{ Hz} \cdot \text{s}^{-1}$

Jede Komponente des Roboteroutputs \underline{p}_{scp0} wurde mit jeweils angepassten Parametern separat angesteuert und die Antworten \underline{s} am Endeffektor gemessen. Wichtig ist anzumerken, dass der Roboter das Chirpsignal nicht ganz korrekt ausführen kann. Einerseits ist die interne Dynamikkompensation nicht optimal, andererseits kommt es bei höheren Frequenzen (ab etwa 4,00-5,00 Hz) zu undefinierten Getriebschwingungen. Hauptauswirkung ist, dass die Amplitude der Anregung mit steigender Frequenz abnimmt (\rightarrow Tiefpassverhalten). Man kann dem in gewissem Rahmen durch Anpassung von $\$FILTER$ begegnen. Jedoch kommen so zusätzlich Dreckeffekte in die Messung. Um jetzt aus \underline{s} den rein-dynamischen Anteil zu extrahieren muss Gleichung (46) wie folgt umgeformt werden:

$$\underline{s}^{dyn}(t) = \underbrace{\underline{s}(t, \underline{p}_{scp0})}_{\text{gemessen}} - \underbrace{\underline{s}^{stat}(\underline{p}_{scp0})}_{\text{errechnet}} \quad (55)$$

Dieser Anteil fließt schließlich in das Optimierungsverfahren ein.

Optimierungsverfahren

Ein einfaches und leicht fassbares Verfahren zur Anpassung ist die Bestimmung der Parameter durch die *Methode der kleinsten Fehlerquadrate* (engl. „*least mean squares*“, kurz LMS). Die grundlegende Idee lässt sich bspw. in [27] nachlesen. Professionelle Mathematikprogramme wie z.B. MATLAB© beinhalten solche Methoden. Liegt für jeden Freiheitsgrad das (tatsächliche) Anregungssignal (s. oben) und das Antwortsignal von jedem Freiheitsgrad vor (vgl. Gleichung (55)), so kann aus diesen beiden auf verschiedene Weise eine Übertragungsfunktion hergeleitet werden. Hier wurde folgender Ansatz verfolgt: Über den Amplitudengang, der durch Division der FOURIER-Transformierten von Ein- und Ausgangssignal errechenbar ist, können die essentiellen Größen der zugrundeliegenden Übertragungsfunktion abgeleitet werden. Der Amplitudengang ist gegeben durch [28]:

$$\left| G_{m \rightarrow n}^{mess}(jf) \right| = \left| G_{m \rightarrow n}^{mess}(f) \right| = \left| \frac{y_n(jf)}{u_m(jf)} \right| = \left| \frac{\mathcal{F}\{y_n(t)\}}{\mathcal{F}\{u_m(t)\}} \right| \quad (56)$$

Mit $\mathcal{F}\{\cdot\}$ dem FOURIER-Operator und j der komplexen Einheit. Die Transformation kann bspw. durch eine in MATLAB© integrierte schnelle FOURIER-Transformation (FFT, engl. *fast FOURIER transformation*) durchgeführt werden.

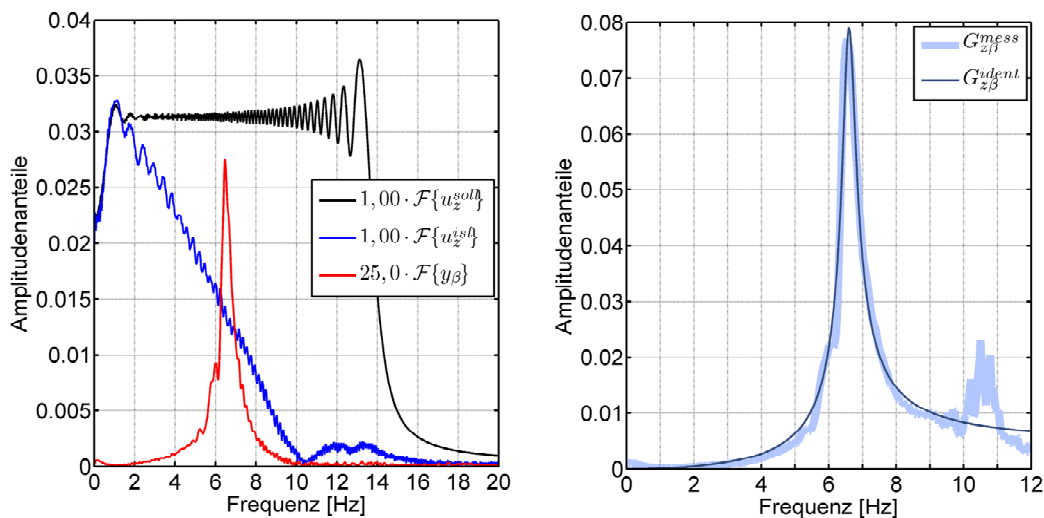


Abbildung 27: Gemessene FFT-Daten¹ zur Identifikation der Übertragungsfunktion $z \rightarrow \beta$ (K_1 ; $K_{z\beta} = 4,73 \cdot 10^{-3}$; $F_\beta = 6,59$ Hz; $D_\beta = 29,5 \cdot 10^{-3}$)

¹ Die „Zacken“ kommen durch mangelhafte Fensterung, also durch Leckeffekte, sind aber unerheblich

In der vorangegangenen Abbildung links sind folgende FFT-Diagramme gezeigt:

- **schwarz**: das Chirpsignal $\mathcal{F}\{u^{soll}\}$ (p^{soll}_{scp0} ; entspricht „SOLL“ in rahmen)
- **blau**: das ausgeführte Chirpsignal $\mathcal{F}\{u^{ist}\} = \mathcal{U}_z(p^{ist}_{scp0}, \text{„MESS“})$
- **rot**: die 25-fach skalierte Antwort $\mathcal{F}\{y\} = \mathcal{Y}_\beta(p_{scp0, scp} \text{ Schwingung des EE})$

Rechts ist der gemessene (G^{mess} , blassblau dick) nach Gleichung (56) und der identifizierte Amplitudengang (G^{ident} , blau) gezeigt.

In [28] findet sich der Rohamplitudengang in Abhängigkeit der Eingangsfrequenz f eines PT2-Systems zu:

$$|G_{mn}(f)| = K_{mn} \frac{f^2}{\sqrt{(f^2 - F_n^2)^2 + (2 \cdot D_n \cdot F_n \cdot f)^2}} \quad (57)$$

Hier sind K_{mn} der stationäre Übertragungsfaktor von $m \rightarrow n$, F_n die Eigenfrequenz des Freiheitsgrades n und D_n dessen Dämpfung (LEHRsches Dämpfungsmaß). Unter Zuhilfenahme der MATLAB® Funktion `lsqcurvefit` wurden nun K_{mn} , F_n und D_n für jeden Freiheitsgrad¹ so angepasst, dass sich gemessener und identifizierter Amplitudengang bestmöglich gleichen. Dies ist in Abbildung 27 rechts als blaue gegenüber der blassblauen Kurve dargestellt. Ebenso nach [28] ergeben sich nun $M \cdot M = 36$ kontinuierliche Übertragungsfunktionen² $G_{mn}(s)$, die wiederum über den Befehl `c2d()` diskretisiert werden können. Siehe dazu auch die MATLAB-Dokumentation in z.B. [29].

Ergebnisdiskussion (→ Details s. Anhang D)

Die Güte dieses Vorgehens variiert von Freiheitsgrad zu Freiheitsgrad. In Abbildung 28 ist eine ($z \rightarrow \beta$) tatsächlich gemessene der identifizierten Ausgangsschwingung aus Abbildung 27 gegenübergestellt. Man sieht, dass sich die identifizierte Schwingung sehr gut mit der gemessenen deckt.

¹ Man sieht, dass der Nenner der Übertragungsfunktionen nur vom Ausgang n abhängt – damit wird die Annahme von Seite 48 untermauert

² Tatsächlich sind 20 $K_{mn}=0$ (ungekoppelt), womit nur 16 Übertragungsfunktionen übrig bleiben

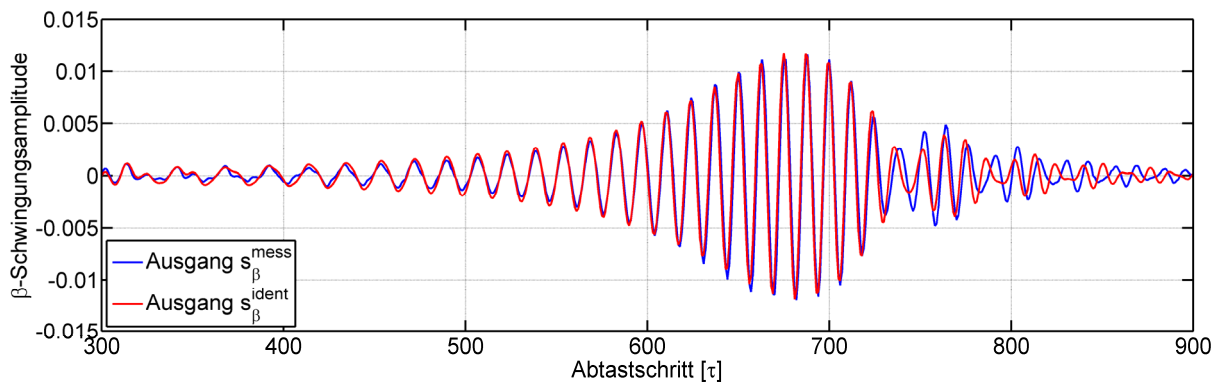


Abbildung 28: Gemessene und identifizierte Systemantwort für ein Chripsignal $z \rightarrow \beta$ im Zeitbereich (vgl. Abbildung 27) (**K1**)

Wendet man das gefundene Modell zur Verifikation auf eine beliebige andere Bahn an, beispielsweise auf die impulse-Bahn, die auch für die Experimente verwendet wurde, zeigt sich folgendes Bild:

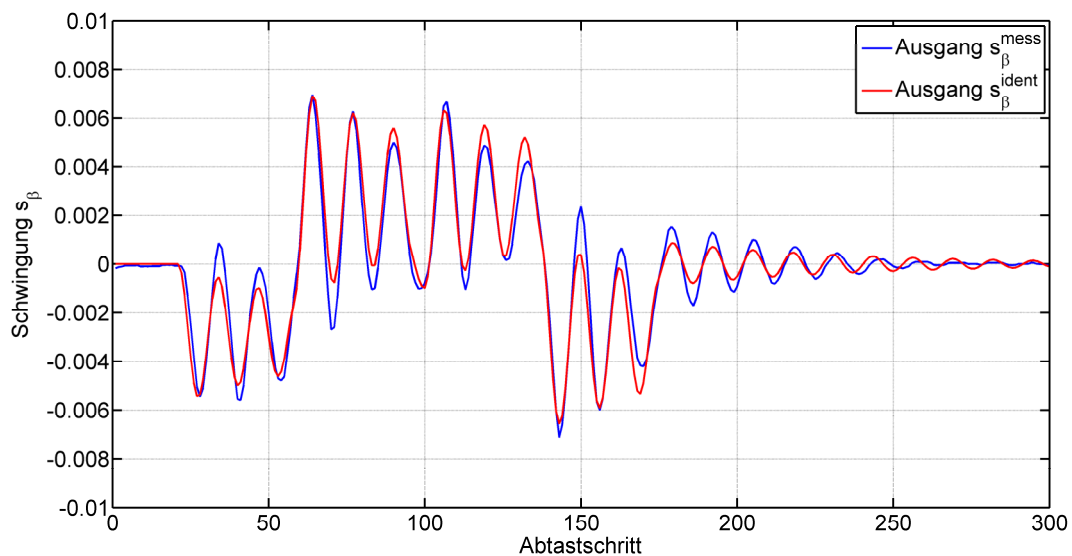


Abbildung 29: Gemessene und identifizierte Systemantwort für die Bahn *impulse10.bahn* $z \rightarrow \beta$ (**K1**, $\$FILTER=0$)

Dieser Weg der Identifikation hat sich als sehr vorteilhaft erwiesen, auch wenn nicht alle Freiheitsgrade ein so gutes Identifikationsergebnis boten wie das Gezeigte. Für verschiedene Freiheitsgrade (insb. x bei **K1** und **K2**) kam es im Roboter zu Resonanzeffekten, womit keine saubere Anregung stattfinden konnte. Man erhielt keine geeignete Kurve, für die das Optimierungsproblem lösbar ist. Es müssen dann andere Methoden verwendet oder Abschätzungen gemacht werden. Zum Beispiel

kann versucht werden, diesen Freiheitsgrad über einen *gekoppelten* Freiheitsgrad anzuregen, für den der Roboter weniger Resonanzeffekte zeigt. Dann können F_n und D_n über den gekoppelten Freiheitsgrad identifiziert werden.

Ein Vorteil dieser Identifikationsmethode ist aber unter anderem, dass hochfrequente „Dreckeffekte“ (bspw. Roboter- und/oder Getriebeeigenschwingungen jenseits der Eigenfrequenzen) nicht mitidentifiziert werden müssen (man begrenzt den Frequenzbereich, in dem die Kurve eingepasst wird; in Abbildung 27 also Frequenzen 0,00-12,00 Hz). Zu achten ist darauf, dass in den FFT-Diagrammen keine „Phantomüberhöhungen“ dort auftreten, wo $\mathcal{F}\{u\} \approx 0$ (\leftrightarrow Division durch eine kleine Zahl in (56); zu Erahnen in Abbildung 27/rechts im Bereich 10-11Hz). Diese werden dann vom Optimierungsalgorithmus unter Umständen als Eigenfrequenzüberhöhung betrachtet.

4.1.4 Gesamte Übertragungsfunktion

In den vorherigen Kapiteln wurden einzelne Komponenten der Gesamtsystemübertragungsfunktion $\mathbf{H}(z)$ hergeleitet, die nun zusammengefügt werden. Werden die Erkenntnisse aus den Gleichungen (45) und (50) in Gleichung (41) eingesetzt, so erhält man mit $\underline{p}_{scp0,scp} = \underline{s}$:

$$\begin{aligned} \underline{p}_{0,scp}^{ist} &= \underline{p}_{0,scp0}^{ist} + \mathbf{J}_0 \cdot \underline{p}_{0,scp0}^{ist} + \mathbf{E}_{dyn}(z) * \underline{p}_{0,scp0}^{ist} + \underline{p}_{scp0,scp}^0 \\ &= \underbrace{(\mathbf{1} + \mathbf{J}_0 + \mathbf{E}_{dyn}(z))}_{\mathbf{H}(z)} * \underline{p}_{0,scp0}^{ist} + \underline{p}_{scp0,scp}^0 \end{aligned} \quad (58)$$

Verwendet man nun noch die Erkenntnis aus (36), so ergibt sich die gesamte Übertragungsstrecke zu:

$$\underline{p}_{0,scp}^{ist} = \underbrace{\frac{\mathbf{E}(z)}{(\mathbf{1} + \mathbf{J}_0 + \mathbf{E}_{dyn}(z))}}_{\mathbf{H}(z)} \cdot \mathbf{R} * \underline{p}_{0,scp0}^{soll} + \underline{p}_{scp0,scp}^0 \quad (59)$$

Mit $\mathbf{E}(z)$ dem gesamten Endeffektor, $\mathbf{R}(z)$ dem Roboter und $\mathbf{H}(z)$ dem gesamten Systemmodell. Die ist auch in folgender Abbildung als Signalflußdiagramm verdeutlicht. Problematisch ist der statische Anteil 0. Ordnung, der nicht mit in die Übertragungsfunktion eingebunden werden kann. Er wird im nächsten Kapitel gesondert behandelt.

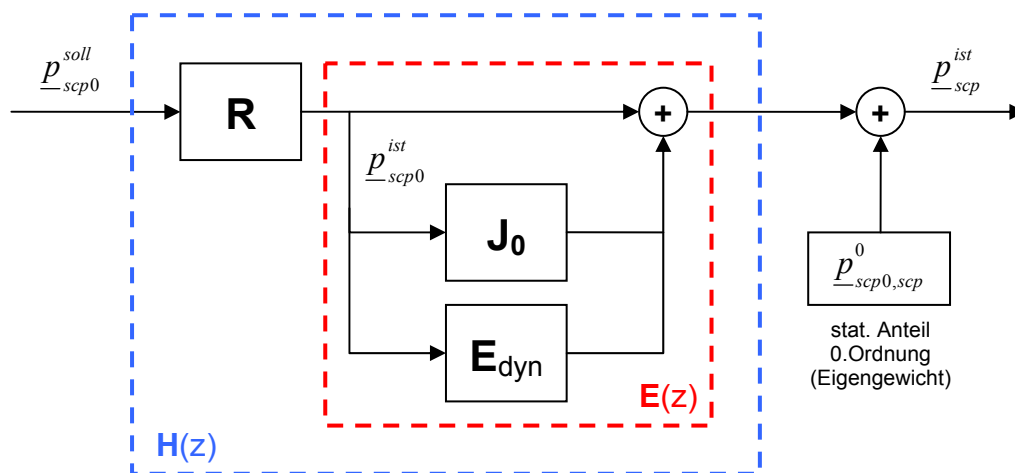


Abbildung 30: Signalflußplan des gesamten Systemmodells $H(z)$

4.2 Regelungsansatz

Mit den Erkenntnissen aus Kapitel 4.1 lässt sich nun die in Kapitel 3 gefundene Vorsteuerung konkret auf das Problem der Endeffektorregelung anwenden. Wichtig hierbei ist, nicht zu vergessen, dass es sich bei $H(z)$ um ein *Modell* der Strecke handelt, welches niemals die Wirklichkeit allumfassend wiedergeben kann. Um diesem Problem wenigstens ansatzweise zu begegnen, wurde neben der reinen Vorsteuerung auch ein Integralregler (Abschnitt 4.2.2) implementiert, der für stationäre Genauigkeit sorgt und somit kleine Modellunsicherheiten ausregeln kann.

4.2.1 Vorsteuerung

Wie im Signalflußplan in Abbildung 30 zu sehen ist, existiert dort der statische Anteil 0. Ordnung, der nicht in die Übertragungsfunktion miteinbezogen werden kann. Um diesen „Versatz“ trotzdem berücksichtigen zu können, muss er vom Sollsignal \underline{w} abgezogen werden, wie Abbildung 31 verdeutlicht. Wichtig ist dabei, dass vorher schon die Transformation des Sollsignals, das bezogen auf den TCP vorliegt, entsprechend der in Kapitel 4.1 gemachten Annahmen auf den SCP zurückgerechnet wurde.

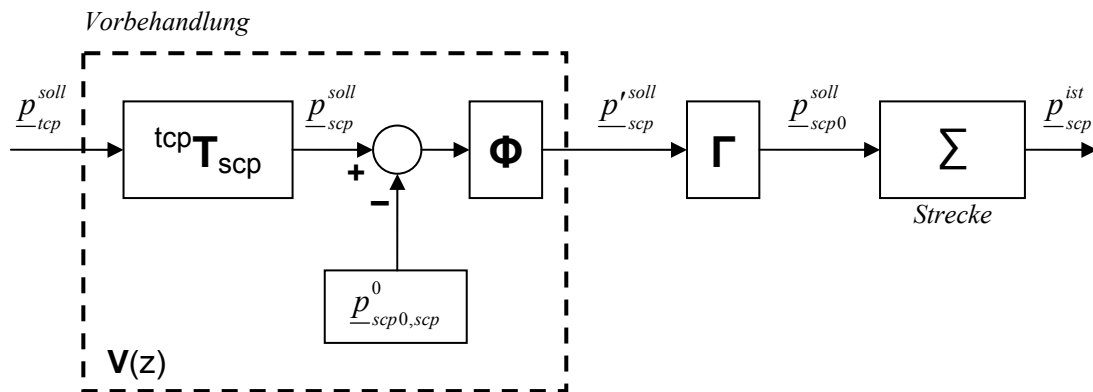


Abbildung 31: Signalflussplan des geschlossenen Regelkreises mit Vorbehandlungsbaustein (Φ für stat. Genauigkeit → Kap. 3.2.4)

Mit dieser Vorbereitung kann nun die Vorsteuerung $\Gamma(z)$ gefunden werden. Zur Realisierung der Regelstruktur, wie in Gleichung (20) gezeigt ist, sind folgende Matrizen und Parameter nötig:

- Die Systemmatrizen C_+ , C sowie \mathcal{D} aus \mathcal{A}_+ , \mathcal{A} , \mathcal{B}_+ und \mathcal{B} .
- Die Strafmatrizen \mathbf{Q} und \mathbf{R}
- Die Horizonte N_p und N_c

Anfangs müssen also die Koeffizientenmatrizen $\mathbf{A}_0 \dots \mathbf{A}_{N_a}$ und $\mathbf{B}_0 \dots \mathbf{B}_{N_b}$, wie sie in Anhang A definiert sind, aus $\mathbf{H}(z)$ extrahiert werden, um daraus C_+ , C und \mathcal{D} zu errechnen. Hat nicht jede Zeile von $\mathbf{H}(z)$ einen gemeinsamen Nenner, so muss zuvor jede Zeile auf einen solchen gebracht werden. Dazu empfiehlt sich eine Erweiterung jedes Elements um die fehlenden Nenner. Als nachteilig ist hierbei anzumerken, dass sich dadurch auch die Dimensionen von N_a und N_b vergrößern, was wiederum zu größeren Regelmatrizen führt. Dies kann ein Nachteil bei der numerischen Genauigkeit, sowie bei der Rechenzeit sein.

Die Matrizen \mathbf{Q} und \mathbf{R} sowie die Horizonte sind wichtige Abstimmungsparameter und müssen während der Einstellungsphase angepasst werden. Ein wesentlicher Punkt bei der Wahl von N_p ist jedoch, dass das Sollsignal \underline{w}_+ für mindestens N_p Zeitschritte zur Verfügung stehen muss. Tut es das nicht, kann versucht werden, die fehlenden Schritte aus den Vorhandenen zu extrapolieren. Sind alle Matrizen vorhanden, so können die Regelmatrizen nach Gleichung (20) bestimmt werden. Diese Darstellungsform hat jedoch gewisse Nachteile bei der Implementierung. Insgesamt

sind in (20) 5 Matrix-Vektormultiplikationen und 3 Summierungen nötig. Sie wird deshalb zur Implementierung so umgeformt, dass sich folgende Struktur ergibt:

$$\underline{U}_+ = \mathcal{K}_w \cdot \underline{W}_+ + \mathcal{K}_y \cdot \underline{Y}_- + \mathcal{K}_u \cdot \underline{U}_- \quad (60)$$

Hierbei errechnen sich die drei neuen Matrizen wie folgt:

$$\begin{aligned} \mathcal{K}_w &= +\mathbf{K} \cdot (\mathbf{G}_+ + \mathbf{R} \cdot \mathbf{1}_{M \cdot N_c \times M \cdot N_p}) \cdot \Phi \\ \mathcal{K}_y &= +\mathbf{K} \cdot \mathbf{G}_+ \cdot \mathcal{D} \\ \mathcal{K}_u &= -\mathbf{K} \cdot \mathbf{G}_+ \cdot \mathcal{C} \end{aligned} \quad (61)$$

Hier ist zu sehen, dass die Korrekturmatrix Φ aus dem Vorbehandlungsblock $\mathbf{V}(z)$ extrahiert und \mathcal{K}_w zugeschlagen wird. $\mathbf{1}_{M \cdot N_c \times M \cdot N_p}$ ist eine Matrix der angegebenen Größe mit Einsen auf der Hauptdiagonalen.

4.2.2 Integralregler

Kein Modell wird die Realität allumfassend wiedergeben können. Zur Herleitung des Regelgesetzes wurden viele Annahmen getroffen, und die Realität wurde unter verschiedenen Vereinfachungen betrachtet. Insbesondere ist die Annäherung der statischen Anteile durch eine TAYLOR-Reihe eine Vereinfachung.

Diesem Problem kann man in Grenzen durch die Anwendung eines integrierenden Rückführkreises begegnen. Durch die Wahl eines kleinen Verstärkungsfaktors \mathbf{K}_I lässt sich der dynamische Einfluss des I-Reglers minimieren und Stabilität gewährleisten. Im diskreten Fall errechnet sich das integrierende Regelgesetz nach folgender Gleichung. Zur einfacheren Implementierung wird die Summe aufgeteilt in einen vergangenen Fehleranteil $\underline{E}_-[k]$ und den aktuellen Fehler $\underline{e}[k]$.

$$\underline{u}_I[k] = \mathbf{K}_I \cdot \sum_{j=-\infty}^k (\underline{w}[j] - \underline{y}_{mess}[j]) = \mathbf{K}_I \cdot \left(\underbrace{\sum_{j=-\infty}^{k-1} (\underline{w}[j] - \underline{y}_{mess}[j])}_{:=\underline{E}_-[k]} + (\underline{w}[k] - \underline{y}_{mess}[k]) \right) \quad (62)$$

Dabei ist nicht zu vergessen, dass das vom I-Regler modifizierte Stellsignal $\underline{u}[k]$ auf alle zukünftigen Stellsignale \underline{U}_+ aufaddiert werden muss:

$$\underline{U}_+^I = \underbrace{\begin{bmatrix} \vdots \\ \mathbf{1}_{M \times M} \\ \vdots \end{bmatrix}}_{Nc\text{-mal}} \cdot \underline{u}_I[k] \quad (63)$$

Das endgültige Stellsignal entsteht durch Addition mit dem von der prädiktiven Vorsteuerung erstellen Signal \underline{U}_+^P nach:

$$\underline{U}_+ = \underline{U}_+^P + \underline{U}_+^I \quad (64)$$

Dieses Signal muss auch dem u -Zweig in Gleichung (20) resp. (60) in den prädiktiven Regler, sowie in den Prädiktor nach Kapitel 3.2.3.2 zugeführt werden. Ansonsten weicht das interne Modell immer weiter vom tatsächlichen System ab, und es kann kein gutes Steuerergebnisse mehr erzielt werden.

Es sollte zukünftig jedoch auch der Fall untersucht werden, in dem der I-Anteil nicht mit in die Prädiktion einfließt, um den Einfluss besser bewerten zu können.

4.2.3 Gesamtstrecke

Es ergibt sich folgende Gesamtstruktur samt Vorbehandlung, Vorsteuer, Prädiktor (nach Kapitel 3.2.3.2), Integralrückführung (hier bezeichnet mit I) und Strecke.

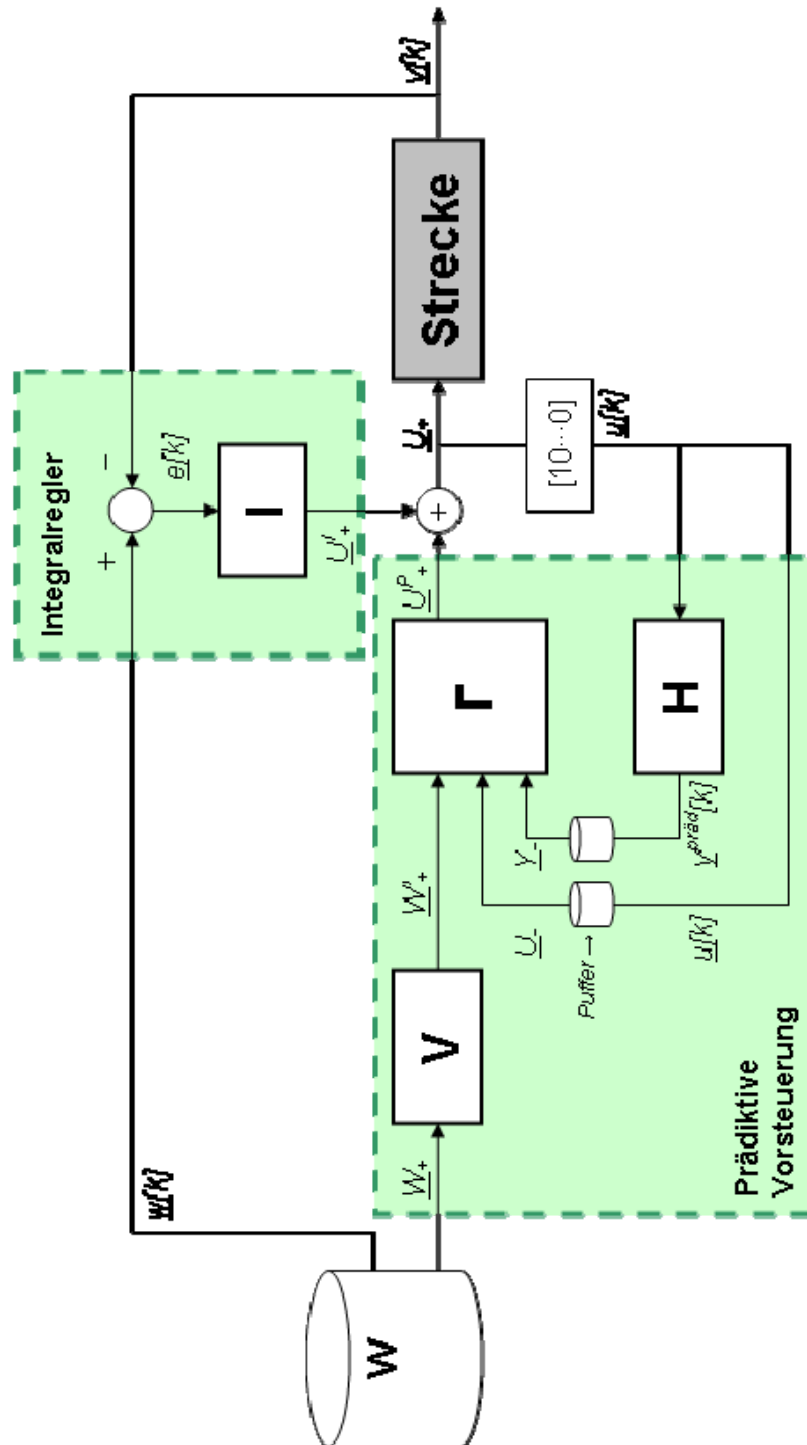


Abbildung 32: Gesamter Signalflossplan

5 Experimente

5.1 Bewertungskriterien

Das Ziel der hier vorgestellten Vorsteuerung ist es, das möglichst *bahntreue* Abfahren einer vorgegebenen Bahn durch die Werkzeugspitze (TCP) zu gewährleisten.

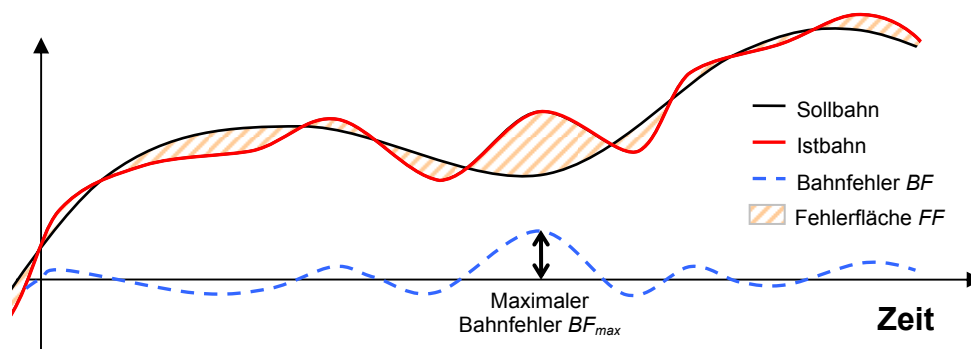


Abbildung 33: Verdeutlichung zweier Kriterien zur Bewertung der Steuergüte (Fehlerfläche FF und max. Bahnfehler BF_{max})

Der Bahnfehler $\underline{BF}[k]$ sei als Differenz zwischen Soll- und Istbahn definiert:

$$\underline{BF}[k] = \underline{p}_{tcp}^{soll}[k] - \underline{p}_{tcp}^{ist}[k] \hat{=} \underline{w}[k] - \underline{y}[k] \quad (65)$$

Aus ihm lassen sich zwei Bewertungskriterien ableiten:

- der max. Bahnfehler \underline{BF}_{max} , der für jeden Freiheitsgrad den Betrag der maximalen Abweichung zwischen Soll- und Istbahn enthält
- die (normierte) Fehlerfläche \underline{FE} als Integral über den Betrag von $\underline{BF}[k]$, normiert mit der Datensatzlänge

Das erste Kriterium bietet eine schnelle und intuitive Möglichkeit, das Steuerergebnis zu bewerten. Ein hoher Wert von \underline{BF}_{max} deutet auf ein schlechtes Steuerergebnis hin. Jedoch kann das Regelergebnis durchaus akzeptabel sein und nur an einer oder weniger Stellen Ausschläge aufweisen. Dann liefert \underline{FE} eine sicherere

Bewertungsmethode, da alle Fehler in die Bewertung mit einfließen. Die Berechnung beider Kriterien ist in der folgenden Gleichung gegeben.

$$\left. \begin{aligned} \underline{BF}_{\max} &= \max\{\|\underline{BF}[k]\|\} \\ \underline{FF} &= \frac{1}{N} \sum_{k=0}^N \|\underline{BF}[k]\| \end{aligned} \right\} \forall k = 0 \dots N \quad (66)$$

Hier sei mit N die Länge des Datensatzes \underline{BF} bezeichnet.

5.2 Experimentenaufbau

Bevor das Regelgesetz am *KUKA KR180* Roboter mit Montageausrüstung am *iwb* in Augsburg getestet werden konnte, wurden alle Annahmen, Identifikationen und Programme an einem *KUKA KR16* am DLR-Standort Oberpfaffenhofen durchgeführt und getestet. Es werden deshalb stets folgende Konfigurationen unterschieden:

#	Roboter	Endeffektor	Bemerkungen
K1	KUKA KR 16-2	8kg Hantelgewichte, ein KMS, keine zusätzlichen Federn	Standort <i>DLR</i> Oberpfaffenhofen, weitgehend lineares Verhalten → siehe auch Abbildung 2
K2	KUKA KR 180	Montageendeffektor, zwei KMS, zusätzliche Federung	Standort <i>iwb</i> Augsburg, nichtlineare Hystereseeffekte → siehe auch Abbildung 1

Tabelle 5: Roboterkonfigurationen

Unter der jeweiligen Konfiguration wurden verschiedene Parametersätze auf verschiedenen Bahnen abgefahren. Dabei war die Bahn mit der Bezeichnung *impulse10.bahn* zentraler Versuchsbestandteil. Die Bahn durchfährt nacheinander parabolische Kurven in jedem Freiheitsgrad (in der Reihenfolge z , y , x und γ , β und α (im SCP-System)). Durch die doppelte Ableitung zur Beschleunigung ergeben sich dadurch Beschleunigungssprünge (Ordnungsreduktion von 2. Ordnung (Parabel) zu 0. Ordnung (Sprünge)). Dies ist in Abbildung 34 exemplarisch für den Freiheitsgrad β gezeigt. Man erkennt an der Beschleunigung (rot) die annähernde Sprungform.

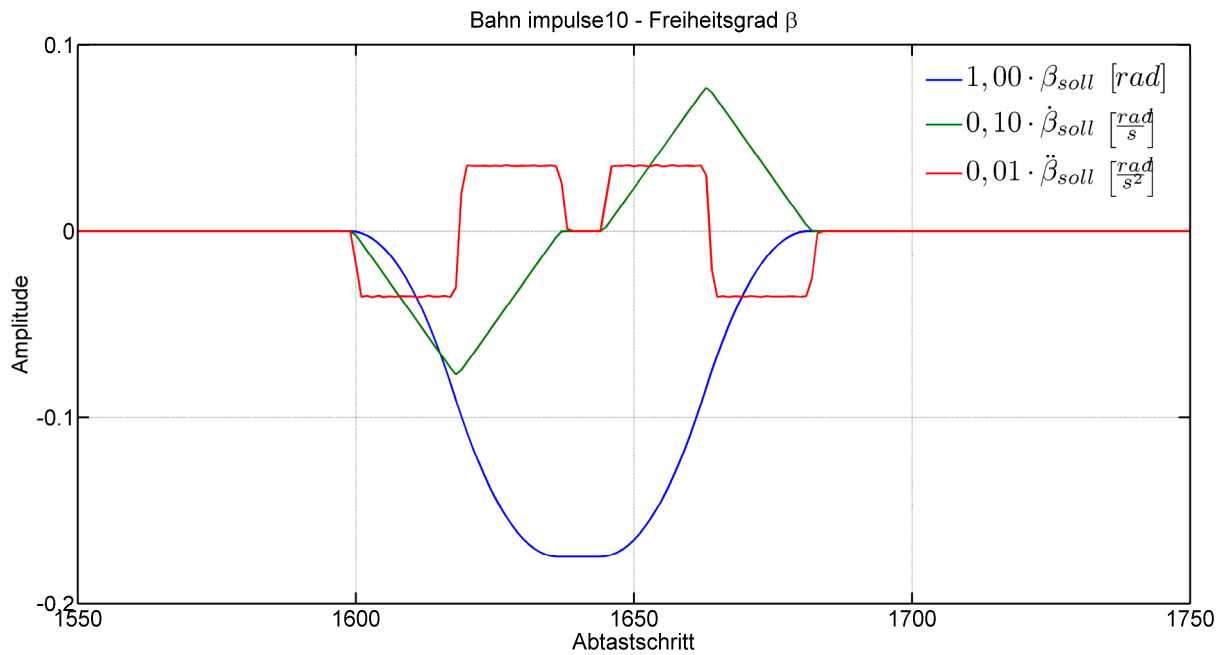


Abbildung 34: Freiheitsgrad β der Bahn impulse10 mitsamt Geschwindigkeits- und Beschleunigungsverlauf (linear bzw. konstant)

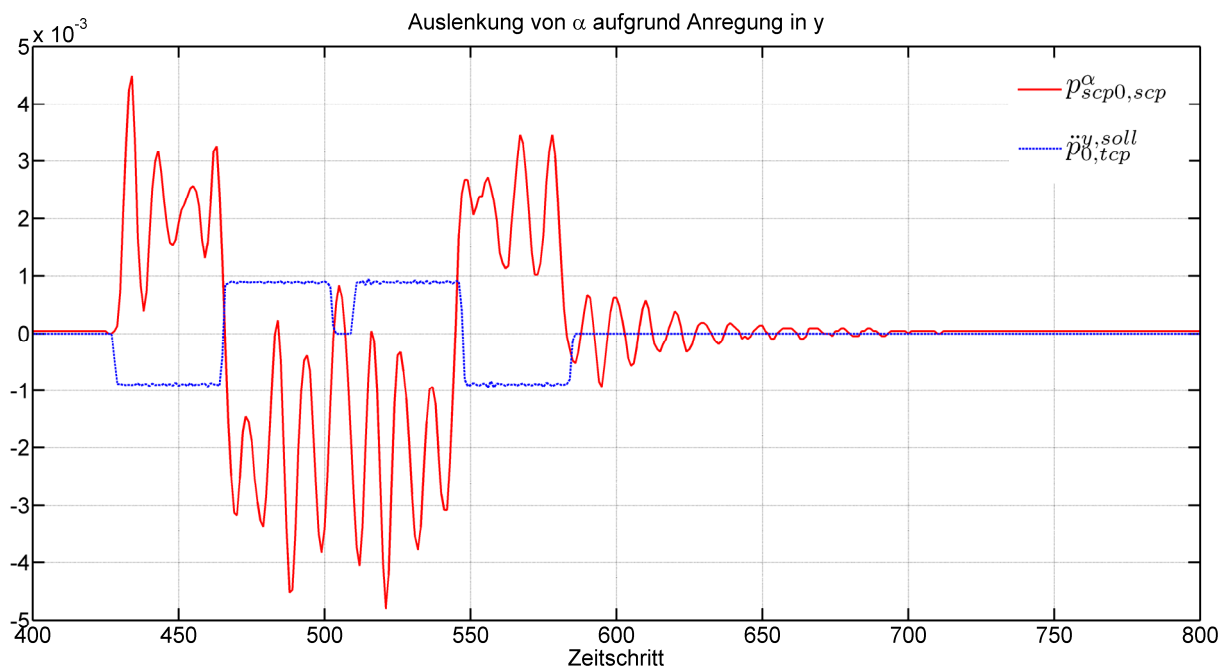


Abbildung 35: Auslenkung von α (rot) aufgrund einer Anregung des gekoppelten Freiheitsgrades y (blau, nicht maßstäblich) für **K1**

Durch das abrupte An-, Um- und Abstellen der Motoren werden die störenden Schwingungen am Endeffektor besonders stark angeregt und somit deutlich in den Messungen sichtbar. Durch Vergleich einer Bahnabfahrt *ohne* Vorsteuerung und *mit* Vorsteuerung kann das Ergebnis der Schwingungsdämpfung gut verglichen werden. In Abbildung 35 ist in rot die Auslenkung des Freiheitsgrades α (Rollwinkel) aufgrund einer Anregung in y-Richtung (Förderrichtung, blau, nicht maßstäblich skaliert) bei ausgeschalteter Vorsteuerung und Konfiguration **K1** gezeigt. Es ist gut zu sehen, dass eine Kopplung (mit negativer Verstärkung) zwischen beiden Freiheitsgraden besteht. Außerdem bestätigt sich offensichtlich die Annahme, dass die dynamischen Auslenkungen des Endeffektors auf die Beschleunigungen am SCP0 zurückzuführen sind. Sichtbar ist auch das Nachschwingen nach Ende der Beschleunigungsphase in der jeweiligen Eigenfrequenz. Die exponentielle Dämpfung lässt sich aber nur erahnen.

Es wurden Experimente mit verschiedenen Parametersätzen, Systemmodellen und Einstellungen am Roboter vorgenommen. Lediglich $N_p=N_c=22$ konnte nicht variiert werden, da dies tiefgreifenderen Eingriff in das bestehende System bedeutet hätte. Insbesondere wurde bei den Systemmodellen untersucht, wie sich die Integration des Mittelwertfilters nach Gleichung (37) auf die Stellgrößen auswirkt.

5.3 Ergebnisse

Wird das Regelgesetz und der Roboter in z.B. MATLAB simuliert, und unterstellt man perfekte Systemkenntnis ($H(z) \equiv \Sigma(z)$), so ist das Regelgesetz in der Lage, Schwingungen sehr gut auszuregeln. Sowohl \underline{BF}_{max} , als auch \underline{FE} gehen gegen 0. Beim Experiment am realen Roboter zeigen sich jedoch einige negative Effekte, die das Regelergebnis verschlechtern. Teilweise ist das Ergebnis *mit* eingeschaltener Steuerung sogar **schlechter** als *ohne* Vorsteuerung.

Wichtig ist anzumerken, dass für jegliche folgende Experimente der Integralregler deaktiviert war, um die Messergebnisse nicht zu verfälschen.

Simulationsergebnisse

Wurde die prädiktive Vorsteuerung am Rechner mit perfekter Systemkenntnis simuliert, so wurden sehr gute Ergebnisse erzielt. In folgender Abbildung ist exemplarisch der Bahnfehler zwischen einem realen Experiment (**K1**, blassrot, Parametersatz 29, s. Anhang E/Tabelle 13) und Simulation (rot) in MATLAB© gezeigt.

Es ist deutlich zu sehen, dass bei der Simulation lediglich am Ende einige wenige Abweichungen im Bereich $\pm 0,01 \text{ rad}$ auftreten. Dagegen häufen sich die Abweichungen beim realen Experimente drastisch.

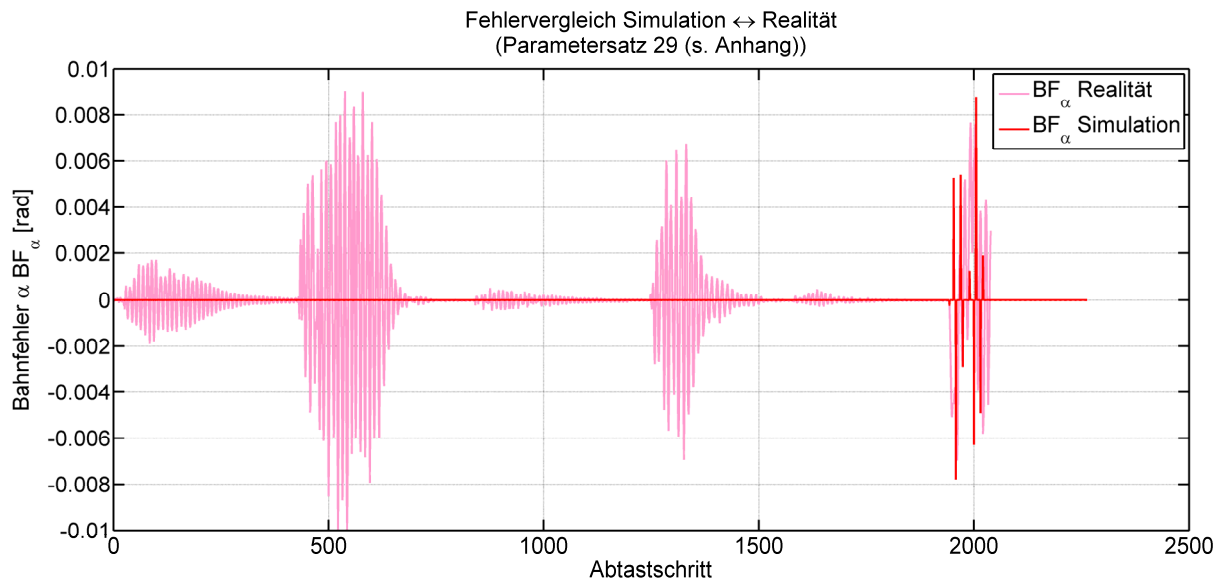


Abbildung 36: Fehlervergleich exemplarisch für α (impulse-Bahn) zwischen Simulation und Realität

Ähnlich gute Ergebnisse konnten für keine realen Experimente erreicht werden.

Aktivierung der Kompensationsvorsteuerung

Wenn man die in Kapitel 4.1.2 angesprochene, schon vor Beginn dieser Arbeit vorhandene, Kompensationsvorsteuerung aktivierte, ergaben sich Probleme. Zwar kompensierte diese Vorsteuerung unter anderem die Totzeit im Roboter, jedoch war ihre genaue Übertragungsfunktion nicht bekannt und konnte somit nicht in das Systemmodell $H(z)$ integriert werden. Durch den Tiefpasscharakter, den diese Vorsteuerung hat, werden „schnelle“ Stellsignale, wie sie von der prädiktiven Vorsteuerung erzeugt werden, abgeschwächt, was zu einem unzureichenden Regelergebnis führt (s. dazu Abbildung 37). Aus diesem Grund wurde die Kompensationsvorsteuerung deaktiviert.

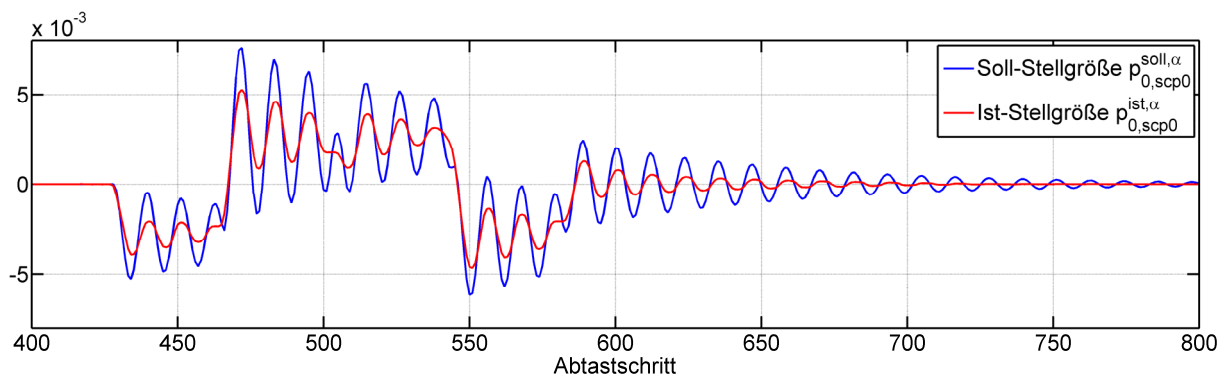


Abbildung 37: Vergleich der Sollstellgröße und der tatsächlich ausgeführten Stellgröße durch Einfluss der Kompensationsvorsteuerung ($\alpha \rightarrow y$)

Einfluss des Mittelwertfilters \$FILTER

Es wurde schon angesprochen, dass das Mittelwertfilter unter gewissen Umständen problematisch sein kann. Berücksichtigte man das Mittelwertfilter *nicht* in der Roboterübertragungsfunktion, ließ es im Roboter jedoch aktiviert ($\$FILTER > 0$), so ergab sich ein ähnlicher Effekt wie im Falle einer aktivierten Kompensationsvorsteuerung. Stellsignalanteile wurden entsprechend dem in Abbildung 23 gezeigten Verhalten je nach Filterbreite abgeschwächt bzw. ganz ausgefiltert.

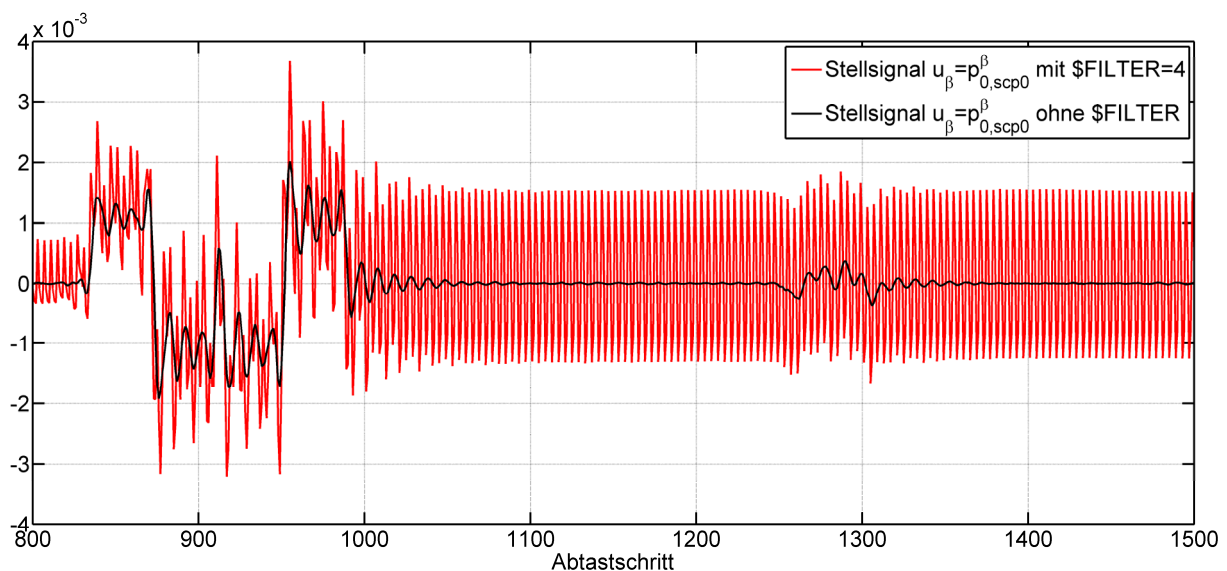


Abbildung 38: Vergleich der Stellgrößen (SCP0-Lage β) zwischen Steuergesetz mit und ohne Berücksichtigung des Mittelwertfilters

Nun ist die Übertragungsfunktion des Mittelwertfilters im Gegensatz zu der der Kompensationsvorsteuerung bekannt (s. Gleichung (37)) und kann somit in der Herleitung des Steuergesetzes berücksichtigt werden. Hierbei hat sich jedoch als nachteilig ergeben, dass durch die *zusätzlichen* nicht-minimalphasigen (ungedämpften) Nullstellen hochfrequente Stellsignale erzeugt werden. Dies ist in Abbildung 38 gezeigt. Zwar werden diese hochfrequenten Anteile (ca. 20Hz) weitgehend von dem Eingangsfiler wieder ausgefiltert. Jedoch bleibt stets ein „Rest“ übrig, der sich als störender Anteil im letztendlich ausgeführten Stellsignal niederschlägt. Die tatsächliche Lage des SCP0 aufgrund des oben gezeigten Eingangssignals ist hier gezeigt:

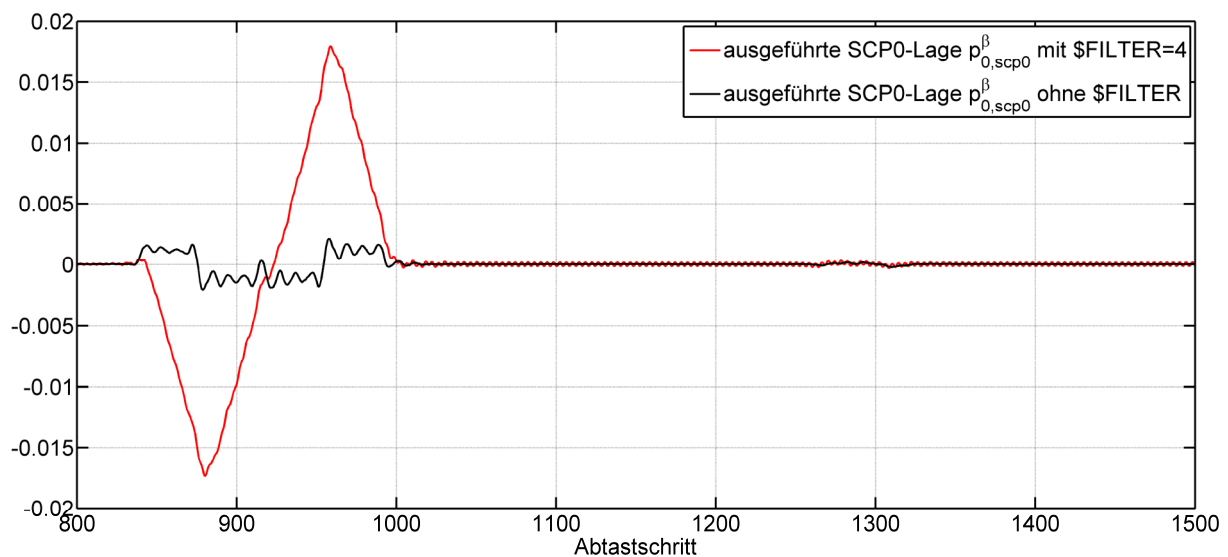


Abbildung 39: Tatsächlich ausgeführte Bewegung am SCP0 bei Eingang wie in Abbildung 38 (für β ; Amplitudengrößenordnung beachten!)

Es ist einerseits sichtbar, dass ein „Zackeneffekt“ auftritt. Der Endeffektor wird in gezeigtem Falle eigentlich in x-Richtung bewegt. Jedoch wird der Nickwinkel des Endeffektors zuerst linear negativ, dann linear positiv verfahren. Nach Schritt 1000 ist auch noch eine Restwelligkeit im roten Signal zu sehen, was sich als Zittern des gesamten Roboters bemerkbar macht. Ohne Mittelwertfilter ist dieser Effekt nicht zu beobachten. Tatsächlich gleichen sich die schwarzen Kurven aus Abbildung 38 und Abbildung 39 fast exakt, nur die Robotertotzeit von ca. 3 Zeitschritten ist bemerkbar.

Da dieser Effekt nur durch eine starke Erhöhung der Stellgrößenstrafen in R reduzieren lässt (was aber gleichzeitig die Effekte der Vorsteuerung überhaupt reduziert), wurde im Weiteren auf das Mittelwertfilter verzichtet.

Experimente ohne Kompensations- und Mittelungsvorsteuerung

Letztendlich mussten die Experimente mit deaktivierter Kompensationsvorsteuerung und ohne Eingangsmittelwertfilter durchgeführt werden. Nur so war die maximale Beherrschbarkeit der Gesamtübertragungsfunktion gegeben. In den folgenden Tabellen sind die Fehler \underline{BF}_{max} und \underline{FF} eines Versuches *ohne* aktivierte prädiktive Vorsteuerung (Spalte „0“, Experiment 101211/2, Anhang E) insgesamt 4 Versuchen *mit* Vorsteuerung (101214/3-6) gegenübergestellt. Dabei wurden die Parametersätze variiert. Zusätzlich angegeben ist, ob sich der Fehler dabei verringert oder erhöht hat.

(a) \underline{BF}_{max}	Parametersatz								
	0	25		26		28		29	
x [mm]	2,409	2,378	-1,29 %	2,403	-0,25 %	2,405	-0,17 %	2,353	-2,32 %
y [mm]	2,904	3,368	15,98 %	3,495	20,35 %	3,435	18,29 %	3,457	19,04 %
z [mm]	2,550	2,425	-4,90 %	2,533	-0,67 %	2,568	0,71 %	2,432	-4,63 %
α [μ m]	11,65	9,591	-17,6 %	10,66	-8,50 %	9,835	-15,6 %	10,54	-9,53 %
β [μ m]	7,185	4,089	-43,09 %	5,161	-28,2 %	4,872	-32,2 %	5,829	-18,9 %
γ [μ m]	6,045	5,806	-3,95 %	5,585	-7,61 %	5,769	-4,57 %	5,219	-13,7 %
durchschnittliche Veränderung:		-13,73 %		-6,21 %		-7,49 %		-8,38 %	

(b) \underline{FF}	Parametersatz								
	0	25		26		28		29	
x [μ m]	163,6	163,3	-0,16 %	166,9	2,03 %	165,4	1,12 %	166,0	1,47 %
y [μ m]	120,6	152,7	26,60 %	178,8	48,23 %	166,6	38,11 %	175,9	45,80 %
z [μ m]	119,1	120,0	0,68 %	120,6	1,23 %	117,6	-1,30 %	118,9	-0,24 %
α [nm]	582,0	744,0	27,84 %	996,0	71,13 %	875,0	50,34 %	942,0	61,86 %
β [nm]	566,0	533,0	-5,83 %	572,0	1,06 %	535,0	-5,48 %	542,0	-4,24 %
γ [nm]	222,0	249,0	12,16 %	279,0	25,68 %	299,0	34,68 %	324,0	45,95 %
durchschnittliche Veränderung:		10,21 %		24,89 %		19,58 %		25,10 %	

Tabelle 6: Veränderung des (a) maximalen Bahnfehlers \underline{BF}_{max} und (b) der max. Fehlerfläche \underline{FF} für 4 Parametersätze

Die verwendeten Parametersätze waren (Auszug aus Anhang E/Tabelle 13):

#	q_n	ε	r_n
25	0 0 0 3 3 3	0,10	0 0 0 0 0 0
26	0 0 0 3 3 3	$R = 0$	
28	0 0 0 5 5 6	$R = 0$	
29	0 0 0 5 5 6	0,10	0 0 0 1 0 2

Tabelle 7: Parametersätze der Experimente aus Tabelle 6

Es fällt auf, dass die maximalen Bahnfehler für jeden Freiheitsgrad außer y verbessert werden können. Im Falle von β ist sogar eine Verbesserung um über 40% möglich. Auf der anderen Seite wird der Flächenfehler in (fast) jedem Fall schlechter. Dies deckt sich mit der Beobachtung. Es war oft erkennbar, dass der maximale Fehler zwar geringer ausfiel, durch eine unzureichende oder sogar verstärkte Schwingungsanregung aber mehr „Fläche“ vorhanden war. Beispielhaft sind für den Parametersatz 25 die Schwingungen am TCP des Freiheitsgrads β (rot) der Schwingung ohne Regelung gegenübergestellt (dargestellt sind nur die Anregungen durch z , x und y in dieser Reihenfolge).

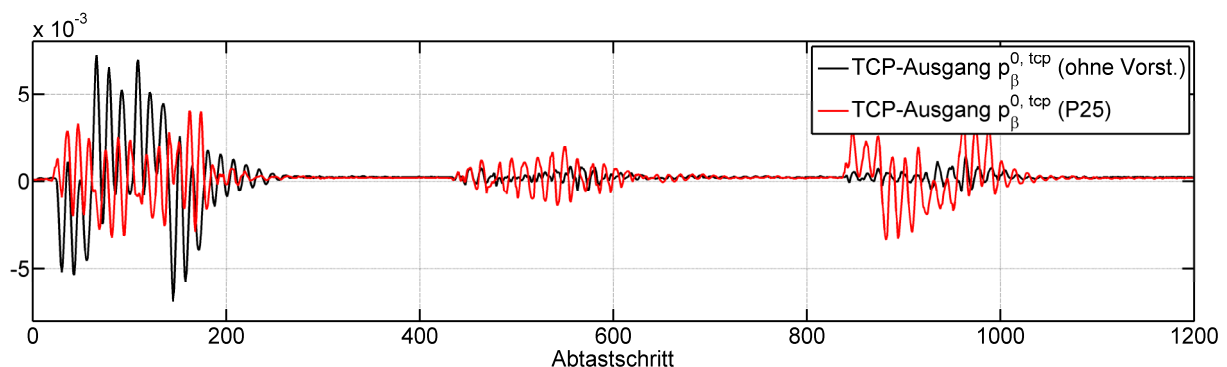


Abbildung 40: TCP-Schwingungen für β mit und ohne prädiktive Steuerung

Der maximale Fehler wird offensichtlich wirklich reduziert (43,09% bei $z \rightarrow \beta$). Die Fehlerfläche ist jedoch größer, was vor allem bei der Anregung durch x im Bereich $t=800-1100$ deutlich wird. Das Steuergesetz produziert ein zu großes Stellsignal, welches nicht mit der Realität korrespondiert. Dadurch wird ein zusätzlicher Fehler erzeugt. Hinzu kommt, dass auch bei der Anregung aus z der Maximalfehler zwar reduziert, die Schwingung aber nicht *ausgedämpft* wird. Lediglich die sprungförmigen Anteile, wie sie schon aus Abbildung 34 und Abbildung 35 bekannt sind, werden ausgeglichen.

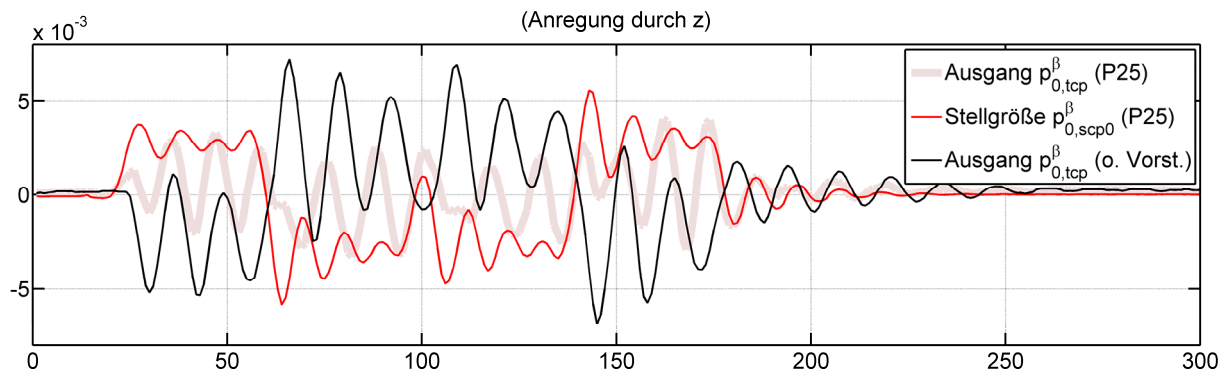


Abbildung 41: Stellgröße und Ausgang für β

Betrachtet man sich diesen Abschnitt (Anregung durch z , $t=0-300$) genauer (Abbildung 41), so erfährt man etwas über die Gründe. Schwarz ist wieder die Schwingung β am TCP ohne jede Vorsteuerung, blassrot nun dieselbe Schwingung mit Vorsteuerung (P25). Rot ist die Nickstellung des SCP0 als Stellgröße. Idealerweise würde diese Nickstellung (des SCP0) gerade die Nickschwingung des Endeffektors *ohne* Vorsteuerung kompensieren. Schwarze Kurve und rote Kurve ergäben zusammen ca. 0. Jedoch erkennt man, dass weder die Amplitude, noch die Frequenz korrekt stimmen. Nur der Mittelwert von ca. $\pm 3 \text{ mrad}$ stimmt in beiden Kurven. So ergibt eine Addition eine Schwingung um ca. 0 mrad . Die statischen Anteile werden kompensiert, die Schwingungen selbst nicht (s. blassrote Kurve).

Speist man jedoch das erzeugte Stellsignal nicht in das echte System, sondern in das Modell ein (unterstellt also perfekte Systemkenntnis), so ergibt sich ein sehr gutes Regelergebnis. Beispielhaft ist das für den Freiheitsgrad α bei Anregung durch y gezeigt.

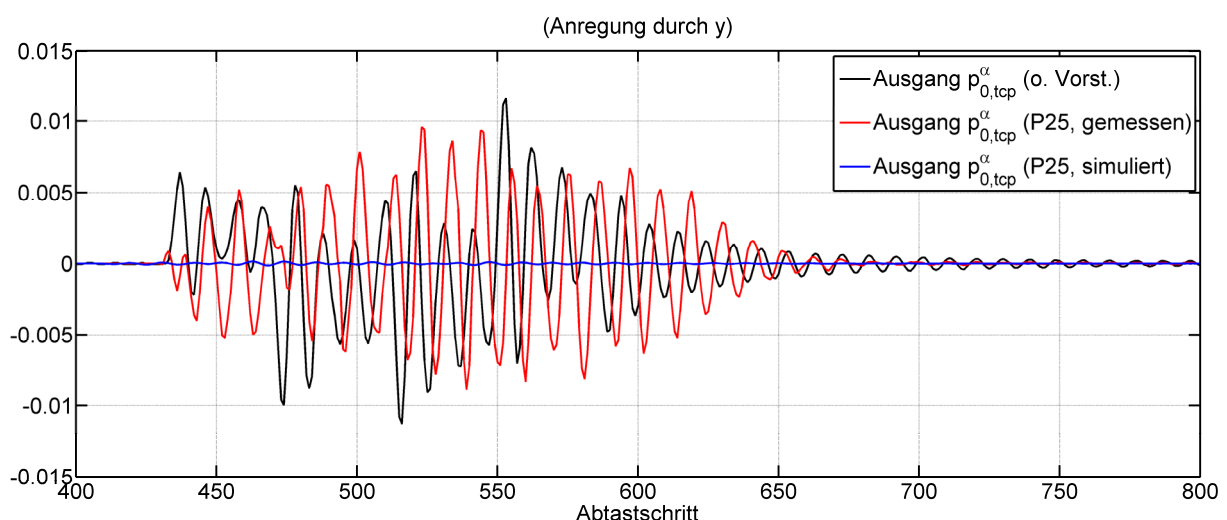


Abbildung 42: Vergleich der Systemausgänge ($y \rightarrow \alpha$); keine Vorsteuerung (schwarz) und Parametersatz 25 (rot gemessen, blau simuliert)

Es existiert nahezu keinerlei Schwingung mehr am α -Ausgang (blaue Kurve). Auch für alle anderen, nicht gezeigten Freiheitsgrade bot sich dieses Bild. Man kann also daraus schließen, dass das mangelhafte Regelergebnis (rote Kurve) in erster Linie auf ein unzureichendes Gesamtmodell zurückzuführen ist.

➔ Vermutlich unzureichendes Systemmodell

Abschließend soll noch ein Blick auf die Kompensation der statischen Gewichtsauslenkung 1. Ordnung geworfen werden (s. Kapitel 4.1.3.1). Dazu wird der Bahnfehler BF_β bei einer Anregung durch β betrachtet. In folgender Abbildung ist dieser für deaktivierte Vorsteuerung (schwarz) und aktivierte Vorsteuerung (rot, P25) gezeigt. Es zeigt sich bei deaktivierter Vorsteuerung deutlich der parabolische Fehler, der aber bei aktivierter Vorsteuerung nicht mehr auftritt.

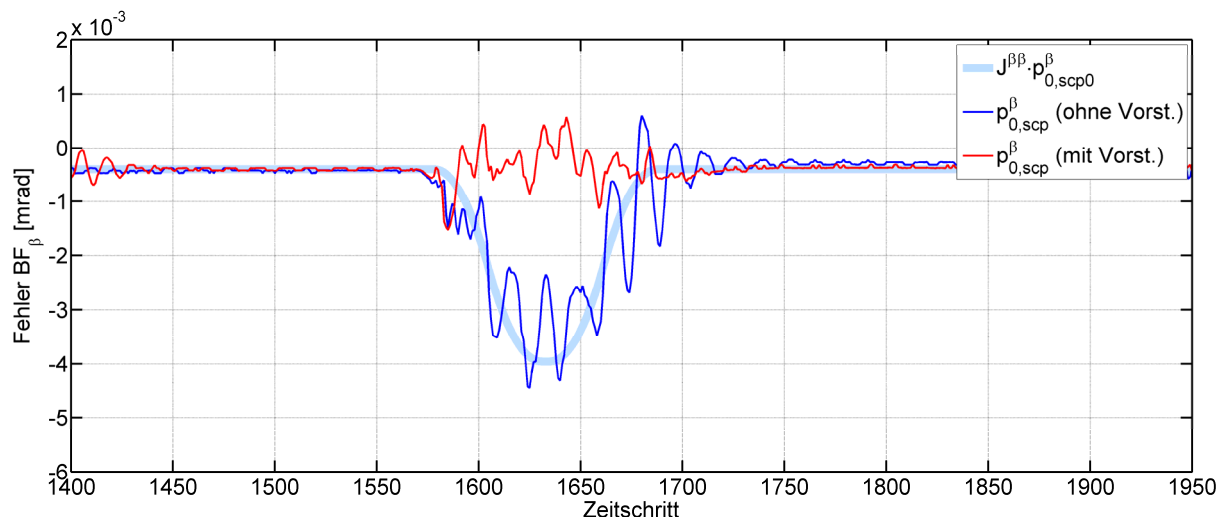


Abbildung 43: Ausgleich des statischen Fehlers 1. Ordnung (für β)

Blassblau ist der unterlagerte Anteil des statischen Fehlers dargestellt.

Einflüsse der Rückkopplung

Während bei der Simulation in MATLAB® die Aktivierung der Rückkopplung nach Kapitel 3.2.3 in der Lage war, Modellunsicherheiten teilweise auszugleichen, führte sie bei allen realen Experimenten zur Instabilität des Systems. Nach jeweils ca. 50-100 Abtastschritten beendete die Robotersteuerung den Ablauf wegen zu hoher Stellmomente der Motoren. Als Grund wird angenommen, dass mit zunehmender Zeit die Phasenabweichung zwischen internem Systemmodell der Steuerung (die auf dem Systemmodell $H(z)$ beruht) und dem realen System $\Sigma(z)$ zu groß wird, und es

schlussendlich zu einer positiven Phasenüberlagerung kommt. Somit wachsen die internen Zustände der Steuerung, das System wird instabil. Auch damit wird die Vermutung bestärkt, dass der Grund für das schlechte Abschneiden der Steuerung ein nicht ausreichendes Systemmodell ist.

Es ist zu erwarten (insb. basierend auf Erkenntnissen aus der Simulation), dass bei besserer Systemkenntnis eine Rückkopplung möglich sein sollte. Dadurch dürfte sich das Regelergebnis verbessern lassen.

Anderes Gütefunktional

Es wurden Experimente gemacht (z.B. Anhang E 101207/2), in denen im Gütefunktional \underline{U}_+ durch eine Annäherung von $\Delta\underline{U}_+$ ersetzt wurde. Dazu wurde wie folgt vorgegangen:

$$\Delta\underline{U}_+ \approx \begin{bmatrix} 1 & \cdots & -1 & & \\ & 1 & \cdots & -1 & \\ & & \ddots & & \ddots \\ & & & 1 & \cdots & -1 \\ & & & & 1 \end{bmatrix} \cdot \underline{U}_+ := \mathbf{D}^1 \cdot \underline{U}_+ \quad (67)$$

Zwischen einer 1 und einer -1 befinden sich $M-2$ Nullen. Es wurde so versucht, nicht den relativen Fehler selbst, sondern dessen erste Ableitung (Relativgeschwindigkeit bzw. Frequenz) zu minimieren. Das Ergebnis war jedoch, dass das Steuersignal fast keine kompensierenden Schwingungen enthielt. Es konnte keinerlei Verbesserung gegenüber einer Gewichtung von \underline{U}_+ festgestellt werden, somit wurden keine weiteren Anstrengungen in dieser Richtung unternommen.

Experimente am *iwb* (K2)

Aufgrund eines Softwarefehlers, der nicht identifiziert und behoben werden konnte, konnten leider keine Experimente am eigentlichen Robotersystem (**K2**) durchgeführt werden. Bei Aktivierung des Regelgesetzes im Steuerungsprogramm brach die interne Robotersteuerung nach ca. 6-7 Takten (<100ms) den Ablauf ab.

Als Fehlerquelle kommt unter Umständen ein Speicherproblem in Frage.

Fazit

Es ist festzuhalten, dass keine wirkliche Kompensation der Schwingungen erreicht wurde. Es konnte zwar der maximale Bahnfehler teilweise nahezu halbiert werden, jedoch wurden die Schwingungen selbst entweder nur ansatzweise gedämpft oder teilweise sogar verstärkt.

Da in der Simulation das Regelergebnis unter der Voraussetzung idealer Systemkenntnis sehr gut ausfiel und da von vornherein der eigentliche Roboter als unbekannt und somit ideal ($R(z)=1$) angenommen werden musste, liegt die Vermutung nahe, dass das mangelhafte Ergebnis in erster Linie auf ein unzureichend modelliertes Gesamtsystem zurückzuführen ist. Insbesondere der Roboter selbst, und dort in erster Linie der Phasengang, wurde als ideal angenommen und müssen für spätere Arbeiten genauer untersucht werden. Auch da die Identifikation der Endeffektordynamik gute Ergebnisse lieferte, muss der Fehler in der Roboterübertragungsfunktion gesucht werden. Ein Aufaddieren von ansonsten gut passenden Schwingungen deutet auf einen Phasenversatz zwischen Realität und Modell hin.

Verbesserungen konnten in der Kompensation der stationären Abweichungen erzielt werden.

6 Diskussion und Ausblick

Es wurde ein mehrdimensionales prädiktives Steuergesetz auf Basis einer Übertragungsfunktionsdarstellung hergeleitet, implementiert und in die Robotersteuerung integriert. Faktoren und Einflussparameter wurden untersucht, das Gesetz wurde am Rechner simuliert und abschließend wurden Experimente zur Schwingungsdämpfung am realen Robotersystem gemacht. Es wurde eine Abstraktion des Systems erstellt und enthaltene unbekannte Parameter und Faktoren identifiziert. Das Systemmodell wurde in Vergleich mit der realen Strecke gestellt und verifiziert. Obwohl das Gesetz in der Simulation sehr gute Ergebnisse lieferte, konnten jedoch am realen System keine Erfolge erzielt werden.

Als Gründe dafür wurden isoliert: Eine modellbasierte Vorsteuerung, die auf einer inversen Strecke basiert, ist generell anfällig für Modellunsicherheiten. Unterstellte man perfekte Systemkenntnis, konnten in der Simulation gute Ergebnisse erzielt werden. Vieles deutet darauf hin, dass das reale System noch zu weit vom angenommenen System abweicht. Obwohl das Systemmodell zumindest für die Endeffektdynamik sehr gut identifiziert werden konnte, mussten andere Übertragungseigenschaften des Systems vernachlässigt oder als ideal angenommen werden. Insbesondere das Übertragungs- und darin vor allem das Phasenverhalten des mechanischen Robotersystems waren unbekannt und stellen mit Sicherheit einen Kernpunkt späterer Arbeiten dar.

Sollte der Ansatz der prädiktiven Steuerung weiter verfolgt werden, müsste der Einfluss der Horizonte N_p und N_c genauer untersucht werden. Außerdem – und dass ist wie angesprochen ein essentieller Punkt – müsste mehr Energie in die genauere Identifizierung des Gesamtsystems gesteckt werden. Insbesondere die Phasenlage schien hier noch nicht ausreichend korrekt abgebildet zu sein. Zudem müsste eine Lösung für die singuläre Darstellung der Lagevektoren, die momentan noch in ein Zwischensystem \mathbf{R} transformiert werden, gefunden werden.

Es ist jedoch trotzdem fraglich, ob ein modellbasierter Vorsteuerungsansatz überhaupt hinreichende Verbesserungen erzielen kann, wenn das Gesamtsystem nicht ausreichend bekannt ist. Vielleicht sollte auf eine Regelung zurückgegriffen werden.

Anhang A: Prädiktive Vorsteuerung

Legt man zugrunde, dass sich ein M -dimensionales ARX-Modell durch folgende Matrizengleichung darstellen lässt (vgl. Gleichung (5))

$$\begin{bmatrix} \mathbf{A}_0 & \mathbf{A}_1 & \cdots & \mathbf{A}_{N_a} \end{bmatrix} \cdot \begin{bmatrix} \underline{y}[k] \\ \vdots \\ \underline{y}[k - N_a] \end{bmatrix} = \begin{bmatrix} \mathbf{B}_0 & \mathbf{B}_1 & \cdots & \mathbf{B}_{N_b} \end{bmatrix} \cdot \begin{bmatrix} \underline{u}[k] \\ \vdots \\ \underline{u}[k - N_b] \end{bmatrix} \quad (68)$$

dann haben die Matrizen $\mathbf{A}_\#$ bzw. $\mathbf{B}_\#$ die folgende Struktur:

$$\mathbf{A}_\# = \begin{bmatrix} a_{1 \rightarrow 1}^\# & a_{2 \rightarrow 1}^\# & \cdots & a_{M \rightarrow 1}^\# \\ a_{1 \rightarrow 2}^\# & a_{2 \rightarrow 2}^\# & & \vdots \\ \vdots & & \ddots & \vdots \\ a_{1 \rightarrow M}^\# & \cdots & \cdots & a_{M \rightarrow M}^\# \end{bmatrix} \in \mathcal{R}^{M \times M} \quad (69)$$

Oft haben die $\mathbf{A}_\#$ s jedoch Diagonalform, d.h. alle Nebenelemente sind 0. Die Struktur für $\mathbf{B}_\#$ ist analog, jedoch haben sie keine Diagonalstruktur. Die Pfeile dienen zur Verdeutlichung der Übertragungsrichtung können weggelassen werden.

Beispiel 2: Zweidimensionales System

Gegeben sei das folgende dynamische System mit $M=2$.

$$\begin{aligned} +5y_1[k] - 3y_1[k-1] &= -1u_1[k-1] + 18u_2[k-1] - 7,5u_2[k-2] \\ -8y_2[k] + 1y_2[k-1] &= +8u_1[k-1] - 2u_2[k-1] + 6u_2[k-2] \end{aligned}$$

Ablesbar sind $N_a=1$ und $N_b=2$. Dann sind die Matrizen nach Gleichung (68) bzw. (69):

$$\begin{aligned} \mathbf{A}_0 &= \begin{bmatrix} 5 & 0 \\ 0 & -8 \end{bmatrix} & \mathbf{A}_1 &= \begin{bmatrix} -3 & 0 \\ 0 & 1 \end{bmatrix} \\ \mathbf{B}_0 &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \mathbf{B}_1 &= \begin{bmatrix} -1 & 18 \\ 8 & -2 \end{bmatrix} & \mathbf{B}_2 &= \begin{bmatrix} 0 & -7,5 \\ 0 & 6 \end{bmatrix} \end{aligned}$$

Aus $\mathbf{A}_\#$ sind nun die Matrizen \mathcal{A}_+ und \mathcal{A}_- wie folgt aufgebaut:

$$\begin{aligned}
 \mathcal{A}_+ &= \begin{bmatrix} \mathbf{A}_0 & & & & \\ \mathbf{A}_1 & \mathbf{A}_0 & & & \mathbf{0} \\ \mathbf{A}_2 & \mathbf{A}_1 & \mathbf{A}_0 & & \\ \vdots & \vdots & \vdots & \ddots & \\ \mathbf{A}_{Np-1} & \mathbf{A}_{Np-2} & \mathbf{A}_{Np-3} & \dots & \mathbf{A}_0 \end{bmatrix} \in \mathcal{R}^{(M \cdot Np) \times (M \cdot Np)} \\
 \mathcal{A}_- &= \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 & \mathbf{A}_3 & \dots & \mathbf{A}_{Na-1} & \mathbf{A}_{Na} \\ \mathbf{A}_2 & \mathbf{A}_3 & \dots & \mathbf{A}_{Na-1} & \mathbf{A}_{Na} & \mathbf{0} \\ \mathbf{A}_3 & \mathbf{A}_4 & \dots & \mathbf{A}_{Na} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ \mathbf{A}_{Np} & \mathbf{A}_{Np+1} & \dots & \mathbf{0} & \dots & \mathbf{0} \end{bmatrix} \in \mathcal{R}^{(M \cdot Np) \times (M \cdot Na)} \quad (70)
 \end{aligned}$$

Die Matrizen \mathcal{B}_+ und \mathcal{B}_- sind aus den $\mathbf{B}_\#$ zusammengesetzt:

$$\begin{aligned}
 \mathcal{B}_+ &= \begin{bmatrix} \mathbf{B}_1 & & & & \\ \mathbf{B}_2 & \mathbf{B}_1 & & & \mathbf{0} \\ \mathbf{B}_3 & \mathbf{B}_2 & \mathbf{B}_1 & & \\ \vdots & \vdots & \vdots & \ddots & \\ \mathbf{B}_{Np} & \mathbf{B}_{Np-1} & \mathbf{B}_{Np-2} & \dots & \mathbf{B}_1 \end{bmatrix} \in \mathcal{R}^{(M \cdot Np) \times (M \cdot Np)} \\
 \mathcal{B}_- &= \begin{bmatrix} \mathbf{B}_2 & \mathbf{B}_3 & \mathbf{B}_4 & \dots & \mathbf{B}_{Nb-1} & \mathbf{B}_{Nb} \\ \mathbf{B}_3 & \mathbf{B}_4 & \dots & \mathbf{B}_{Nb-1} & \mathbf{B}_{Nb} & \mathbf{0} \\ \mathbf{B}_4 & \mathbf{B}_5 & \dots & \mathbf{B}_{Nb} & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ \mathbf{B}_{Np} & \mathbf{B}_{Np+1} & \dots & \mathbf{0} & \dots & \mathbf{0} \end{bmatrix} \in \mathcal{R}^{(M \cdot Np) \times (M \cdot Nb)} \quad (71)
 \end{aligned}$$

Es ist wichtig anzumerken, dass (per Definition) gilt

$$\begin{aligned}
 \mathbf{A}_j &= \mathbf{0} \quad \Leftrightarrow j > N_a \\
 \mathbf{B}_j &= \mathbf{0} \quad \Leftrightarrow j > N_b
 \end{aligned} \quad (72)$$

Das trifft insbesondere auf $j \geq N_p$ zu, da der Prädiktionshorizont meist größer als N_a bzw. N_b gewählt wird. In folgender Abbildung ist eine Übersicht über die Strukturen der Matrizen und der Dimensionen gegeben.

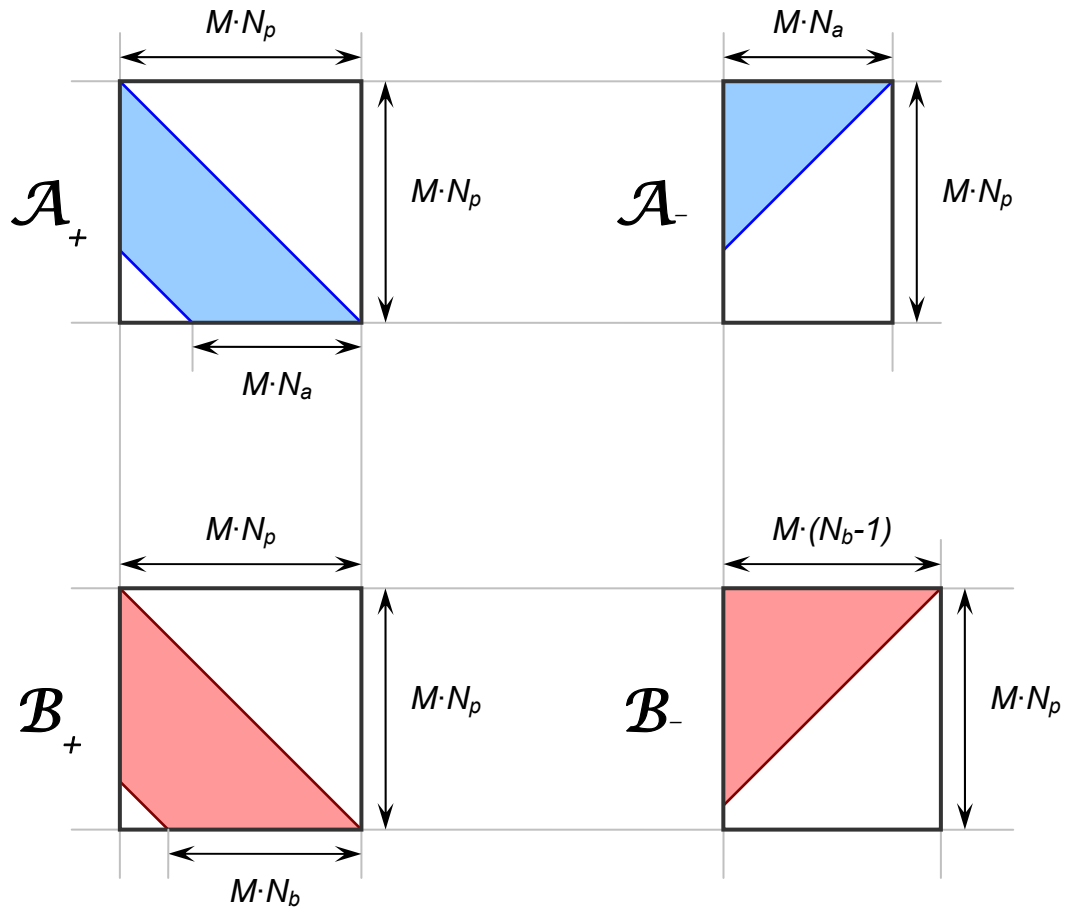


Abbildung 44: Veranschaulichung der Dimensionen und Besetzung der grundlegenden Systemmatrizen

In folgender Tabelle sind weitere Dimensionen zusammengefasst:

Matrix	Zeilen	Spalten
C_+	$M \cdot N_p$	
C	$M \cdot N_p$	$M \cdot (N_b - 1)$
\mathcal{D}	$M \cdot N_p$	$M \cdot N_a$
K	$M \cdot N_c$	
G_+	$M \cdot N_p$	

Tabelle 8: Sonstige Matrixdimensionen

Anhang B: Mathematischer Anhang

Mehrdimensionale z-Transformation

Transformiert man eine skalare, diskret-zeitabhängige Größe x in den Laplace-Bereich, so ergibt sich folgende Form:

$$x[k \pm j] \Rightarrow \mathcal{X} \cdot z^{\pm j} \quad (73)$$

Ist \underline{x} vektoriell und hat die Dimension M , so ergibt sich analog:

$$\underline{x}[k \pm j] \Rightarrow \underline{\mathcal{X}} \cdot z^{\pm j} \quad (74)$$

Für die hier verwendeten Großvektoren der Form

$$\underline{X} = \begin{bmatrix} \underline{x}[k \pm i] \\ \vdots \\ \underline{x}[k \pm j] \end{bmatrix} \in \mathcal{R}^{M \cdot |i-j| \times 1} \quad (75)$$

in denen mehrere Einzelvektoren der Dimension M auftauchen, ergibt sich mit den Gleichungen (74) und (75):

$$\underline{X} \Rightarrow \begin{bmatrix} \underline{\mathcal{X}} \cdot z^{\pm i} \\ \vdots \\ \underline{\mathcal{X}} \cdot z^{\pm j} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{1}_{M \times M} \cdot z^{\pm i} \\ \vdots \\ \mathbf{1}_{M \times M} \cdot z^{\pm j} \end{bmatrix}}_{\zeta_j^i} \cdot \underline{\mathcal{X}} \doteq \zeta_j^i \cdot \underline{\mathcal{X}} \quad (76)$$

Hierbei ist ζ_j^i der mehrdimensionale Zeitverschiebeoperator für die Zeitpunkte $i \dots j$.

Verkürzte Notation der Strafmatriizen

Die in Gleichung (32) eingeführte Notation soll hier kurz genauer erläutert werden. Sei \mathbf{r}^k gegeben durch (analog für \mathbf{q}^k):

$$\mathbf{r}^k = \varepsilon \cdot \begin{bmatrix} \ddots & & 0 \\ & 10^i & \\ 0 & & \ddots \end{bmatrix} = \varepsilon \cdot \underset{i \in \Lambda}{diag}(10^i) \quad (77)$$

Hier bei sei Λ die Menge der M ($=6$) Exponenten, die jedem einzelnen der M Freiheitsgrade zugeordnet ist¹. ε ist ein Skalierungsfaktor. Es wird nun geschrieben:

$$\mathbf{r}^k = \varepsilon \cdot 10^{diag\{\Lambda\}} \quad (78)$$

Beispiel 3: Verkürzte Notation

Gegeben sei das folgende elementare Strafmatrix ($M=3$):

$$\mathbf{q}^k = 0,25 \cdot \begin{bmatrix} 10^3 & 0 & 0 \\ 0 & 10^0 & 0 \\ 0 & 0 & 10^{-2} \end{bmatrix}$$

Dann ist $\varepsilon=0,25$ und $\Lambda=\{9; 0; -2\}$. Verkürzt geschrieben:

$$\mathbf{q}^k = 0,25 \cdot 10^{diag(\{ \ 3; \ 0; \ -2 \ })} \quad (79)$$

¹ Die Wahl von 10 als die Grundzahl ist technisch motiviert – es ist naheliegend, verschiedene Freiheitsgrade durch die Unterscheidung ihrer Größenordnung unterschiedlich zu gewichten.

Anhang C: Robotik

Es existiert sehr umfangreiche Literatur zu diesem Thema. Dieser Anhang soll dazu dienen, kurz relevante Dinge zusammengefasst vorzufinden. Für alles darüber hinaus ist die Literatur zu konsultieren.

Insbesondere wird in diesem Abschnitt auf Konventionen eingegangen, wie sie in *dieser* Arbeit getroffen bzw. verwendet werden. Dazu gehören z.B. Notation und Drehreihenfolgen.

Drehreihenfolge, Singularität

Während die *Position* eines Punktes problemlos durch die Angabe seiner Komponenten entlang der Achsen des darstellenden Koordinatensystems angegeben werden kann, muss zur Beschreibung seiner *Orientierung* jedoch eine Konvention getroffen werden, in welcher Reihenfolge und ab welcher Achsen die drei Raumwinkel gemessen werden. Es existieren unendlich viele Darstellungsformen, jedoch haben sich einige Standardformen etabliert. Hier wird folgende Drehreihenfolge verwendet: Der Winkel α wird als Drehung des Körpers um die z -Achse, der Winkel β um die *neue* y -Achse („Nicken“) und der Winkel γ als Drehung um die neue x -Achse aufgefasst (vgl. Abbildung 45). Es existiert vielseitige Literatur, z. B. in [30].

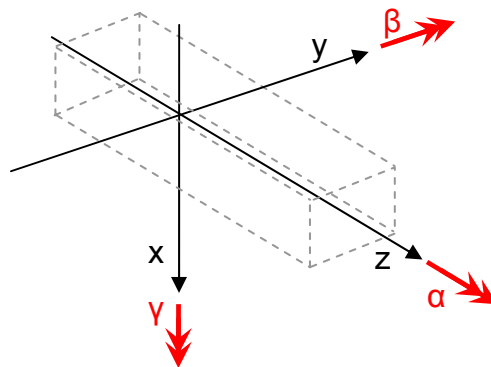


Abbildung 45: Darstellung der Drehreihenfolge
mit α um z , β um y und γ um x

Es resultiert eine Drehmatrix $\mathbf{A}(\alpha, \beta, \gamma)$ nach folgender Form (die genaue Belegung kann in der Literatur gefunden werden und ist zur Verdeutlichung der Singularitätsproblematik unerheblich):

$$\mathbf{A}(\alpha, \beta, \gamma) = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \quad (80)$$

Aus ihr können die drei Winkel rückgerechnet werden:

$$\sin(\beta) = -A_{13} \quad \cos(\alpha) = \frac{A_{11}}{\cos(\beta)} \quad \cos(\gamma) = \frac{A_{33}}{\cos(\beta)} \quad (81)$$

Für $\beta = \pm 90^\circ$ werden die Nenner der beiden letzten Terme 0. Somit lassen sich die Winkel nicht rückrechnen, die Lage ist *singulär*. Die Konsequenz ist, dass die kartesische Lage eines Körpers, dessen Nickwinkel im Bereich $\beta = \pm 90^\circ$ eines Koordinatensystems \mathbf{K} ist, nicht eindeutig in diesem dargestellt werden kann. Die Lage muss in einem singularitätsfreien Koordinatensystem dargestellt werden.

Notation

Die in Kapitel 3.1.1 schon eingeführte Notation der verschiedenen Referenzierungsmöglichkeiten von Punkten soll hier kurz vertieft werden. Die Lagevektoren \underline{p} setzen sich wie folgt zusammen:

$${}_K \underline{p}_{A,B} = \left(\overbrace{\begin{matrix} p_x & p_y & p_z \end{matrix}}^{\text{Lage}} \middle| \overbrace{\begin{matrix} p_\alpha & p_\beta & p_\gamma \end{matrix}}^{\text{Orientierung}} \right)^T \quad (82)$$

Dabei ist \mathbf{K} das darstellende Koordinatensystem, und \underline{p} stellt einen Zusammenhang zwischen Punkt B relativ zu Punkt A her. Die ersten 3 Elemente sind die Positionen \underline{r}_{AB} , gemessen entlang der Beine von \mathbf{K} . Die letzten drei Elemente sind die Orientierungswinkel \underline{o}_{AB} , wie sie weiter oben definiert wurden.

Hilfreich kann noch folgender Zusammenhang sein:

$${}_K \underline{p}_{A,B} = - {}_K \underline{p}_{B,A} \quad (83)$$

Die homogenen Transformationsmatrizen ${}^A\mathbf{T}_B$ haben folgende Struktur:

$${}^A\mathbf{T}_B = \begin{bmatrix} \mathbf{A}_{A \rightarrow B}(\underline{o}_{AB}) & \underline{r}_{AB} \\ 0 & 1 \end{bmatrix} \in \mathcal{R}^{4 \times 4} \quad (84)$$

Oben links befindet sich die Transformationsmatrix aus Gleichung (80), und in den ersten 3 Elementen der 4. Spalte steht der Positionsvektor aus (82). Singularitätsfreiheit vorausgesetzt, ist also eine Umrechnung zwischen Lagevektoren und homogenen Transformationsmatrizen einfach möglich.

Eine Zusammenfassung der Freiheitsgrade und Koordinatensysteme ist nochmals in Abbildung 46 gezeigt.

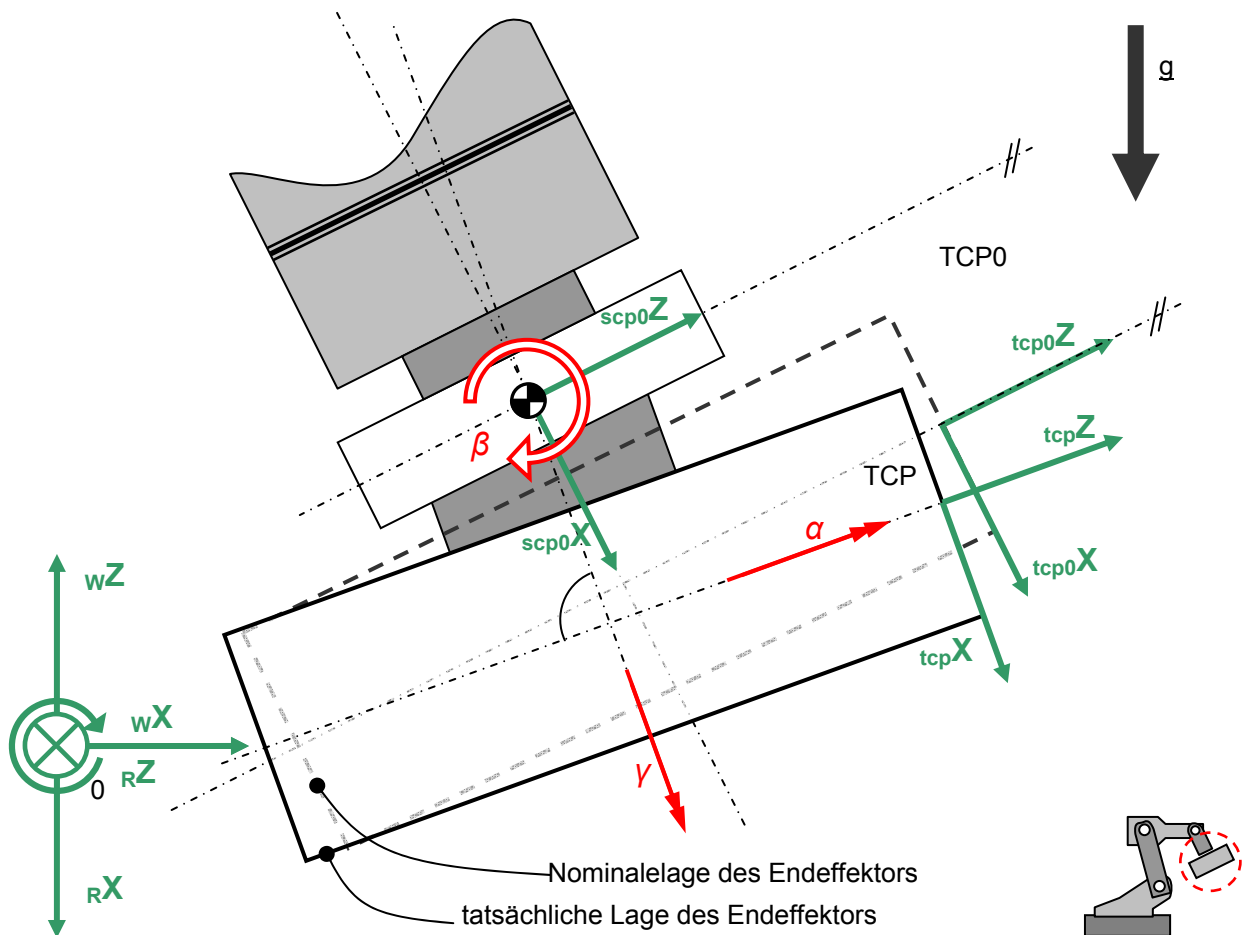


Abbildung 46: Schematische Darstellung des ausgelenkten Endeffektors

Anhang D: Systemidentifikation

Ergebnisse der Endeffektor-Identifikation E_{dyn} für K1

FHG	F_n	D_n	a_i			Bemerkungen
	[Hz]	10^{-3}	a_0	a_1	$a_2=a_{Na}$	
x	10,30	(26,2)	1	-1,3983	0,9603	Dämpfungsparameter nicht genau feststellbar
y	8,15	27,26	1	-1,6070	0,9671	
z	6,52	32,38	1	-1,7356	0,9687	
α	8,18	28,86	1	-1,6032	0,9650	
β	6,51	31,59	1	-1,7370	0,9675	
γ	(7,9)	(36,0)	1	-1,6209	0,9580	Schätzungen (FFT-Ergebnisse mangelhaft)

Tabelle 9: Schwingungsparameter von E_{dyn} für K1

Ein Aus	x	y	z	α	β	γ
x	(-0,16) $\frac{mm}{mm}$		-0,181 $\frac{mm}{mm}$		(-48,1) $\frac{mm}{rad}$	
y		-0,110 $\frac{mm}{mm}$		(+6) $\frac{mm}{rad}$		
z	+7,43m $\frac{mm}{mm}$		-24,3m $\frac{mm}{mm}$		+4,15 $\frac{mm}{rad}$	
α		-6,78m $\frac{rad}{mm}$		(+0,38) $\frac{rad}{rad}$		
β	-17,7m $\frac{rad}{mm}$		+4,64m $\frac{rad}{mm}$		-0,45 $\frac{rad}{rad}$	
γ		-0,85m $\frac{rad}{mm}$		69,57m $\frac{rad}{rad}$		(-0,8) $\frac{rad}{rad}$

Tabelle 10: Übertragungsparameter K_{mn} von E_{dyn} für K1;
 $m=10^{-3}$, eingeklammerte Werte unsicher, $\approx 0,00$

Ergebnisse der Endeffektor-Identifikation E_{dyn} für K2

Die Identifikation für **K2** war weitaus problematischer als für **K1**. Folgende Daten sind Anhaltspunkte und müssen mit Vorsicht genossen werden.

FHG	F_n	D_n	Bemerkungen
	[Hz]	10^{-3}	
x	7,22	21,84	
y	4,16	42,86	
z	(3,61)	(55,80)	
α	(3,85)	(17,84)	
β	(3,46)	(56,10)	
γ	(4,57)	(7,49)	

Tabelle 11: Schwingungsparameter von E_{dyn} für K2

Ein Aus	x	y	z	α	β	γ
	-0.095 $\frac{mm}{mm}$?	?	?	?	?
x		-0,114 $\frac{mm}{mm}$		(-11,1) $\frac{mm}{rad}$		
y			-44,0m $\frac{mm}{mm}$?	
z		-1,42m $\frac{rad}{mm}$		-0,241 $\frac{rad}{rad}$		
α	-0,31m $\frac{rad}{mm}$		+1,25m $\frac{rad}{mm}$		-0,239 $\frac{rad}{rad}$	
β		?		?		-0,29 $\frac{rad}{rad}$
γ						

Tabelle 12: Übertragungsparameter K_{mn} von E_{dyn} für K2;
 $m=10^{-3}$, eingeklammerte Werte unsicher, ≈ 0

Zur Verdeutlichung der Problematik sind hier einige FFT-Plots der Identifikationsexperimente am **K2** gezeigt.

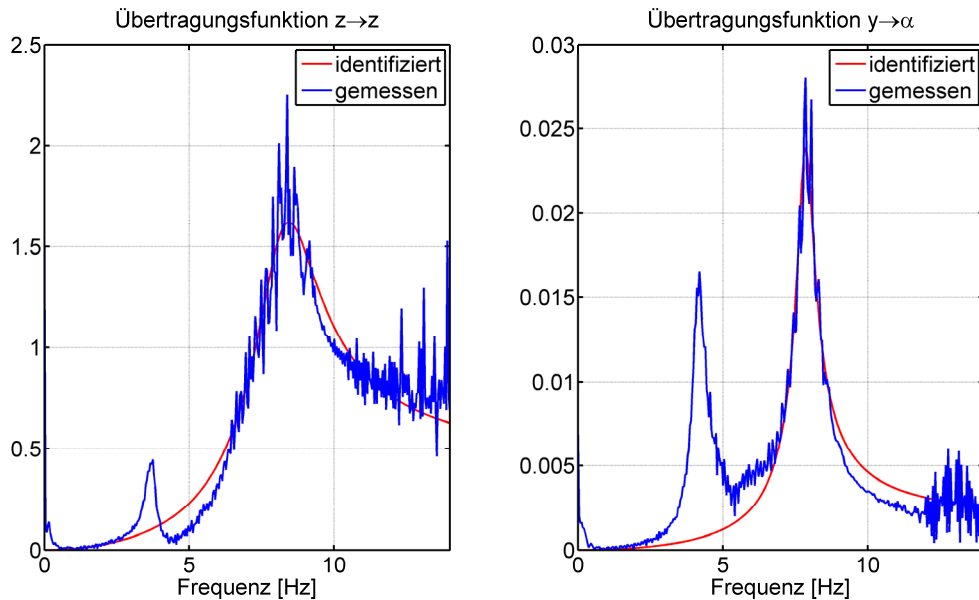


Abbildung 47: Frequenzgang der Übertragungsfunktionen $z \rightarrow z$ (li.) und $y \rightarrow \alpha$ (re.) für Roboter **K2**

Es scheint bei einigen Freiheitsgraden unvorhergesehene Eigenfrequenzen bzw. Resonanzkopplungen zu geben, hier etwa bei ca. 4 Hz.

Ein großes Problem stellte jedoch dar, dass die Identifikation der Kopplungen schwierig verlief. Während für $x \rightarrow x$ eine Frequenz von $F_x \approx 7,22 \text{ Hz}$ identifiziert wurde, ergab $\beta \rightarrow x$ eine Eigenfrequenz $F_x \approx 6,49 \text{ Hz}$. Die Dämpfungen wichen noch weiter von einander ab.

Anhang E: Experimente

In diesem Anhang ist ein Überblick über alle gemachten Experimente gegeben. In der ersten Tabelle sind die Parametersätze definiert.

#	q_n	ϵ	r_n	Bemerkung
1	0 0 0 0 0 0	1,00	0 0 0 0 0 0	basierend auf K1 Systemidentifikation vom Okt. 2010 $N_p=N_c=22$ \$FILTER und Totzeit nicht berücksichtigt
2			0 0 0 2 4 2	
3	2 0 2 0 6 0			
4		0,10	0 3 0 2 4 2	
5	3 3 3 0 0 0	1,00	0 0 0 3 3 0	
6	1 3 3 4 5 0			
7	4 8 8 6 6 3		0 0 0 2 3 0	
8	9 9 9 6 6 3		0 0 0 0 0 0	
9	9 9 9 6 6 6	0,10		
10		0,05		
11	9 9 9 6 6 6	0,10	0 0 0 0 0 0	ab hier Systemidentifikation lt. Anhang D K1
12	9 9 9 6 6 6			Gütefunktional angepasst (s. (67))
13	6 6 6 9 9 9			Gütefunktional wieder normal
14	3 3 3 2 0 2	1,00	0 0 0 5 6 5	K1 , \$FILTER = 8, Totzeit 3
15		10,00		K1 , \$FILTER = 8, Totzeit 3
16			0 0 0 4 4 4	K1 , \$FILTER = 8, Totzeit 3
17	3 3 3 2 0 2	10,00	0 0 0 4 4 4	K2 (→ Anhang D)
18	3 3 3 2 0 2	10,00		K1 , \$FILTER = 8, Totzeit 3
19	existiert nicht			
20	existiert nicht			
21	9 9 9 6 6 6	0,01	0 0 0 0 0 0	K1 , \$FILTER = 0, Totzeit 3
22	0 0 0 0 0 0	0,00	–	K1 , \$FILTER = 0, Totzeit 1
23	9 9 9 6 6 6	0,10	0 0 0 0 0 0	K1 , \$FILTER = 0, Totzeit 1
24	3 3 3 0 0 0	0,10		K1 , \$FILTER = 0, Totzeit 1
25	0 0 0 3 3 3	0,10		K1 , \$FILTER = 0, Totzeit 1
26		0,00	–	K1 , \$FILTER = 0, Totzeit 1
27	existiert nicht			
28	0 0 0 5 5 6	0,00	–	K1 , \$FILTER = 0, Totzeit 1
29		0,10	0 0 0 1 0 2	K1 , \$FILTER = 0, Totzeit 1

Tabelle 13: Überblick über alle Parametersätze (Feld leer→s. Feld darüber)

Liste aller Experimente

#	Rob	P	KV	\$F	PM	I	Bemerkungen
3.11.							
1	K1 r	1	1	8	0	0	
2		1	1	8	99	0	Abbruch
3		2	1	8	0	0	
4		2	1	8	99	0	keine Daten
5		3	1	8	0	0	
6		3	1	8	99	0	keine Daten
7		4	1	8	0	0	
8		4	1	8	99		keine Daten
9.11.							
0	K1 r	--	1	8	--	1	
1		5	1	8	0	1	ohne Endeffektor
2		5	1	8	0	1	ohne Endeffektor
3		4	1	8	0	1	ohne Endeffektor
4		4	0	8	0	1	ohne Endeffektor
5		5	0	8	0	1	ohne Endeffektor
12.11.							
1	K1 r	5	1	8	0	0	neutrainierte Vorsteuerung
2		4	1	8	0	0	
3		6	1	8	0	0	
4		--	0	8	--	0	Bahn: stat_traj_B.bahn
5		--	0	8	--	0	Bahn: stat_traj_B.bahn
6		6	?	8	0	0	
16.11.							
1	K1 r	7	1	8	0	0	
2		8	1	8	0	0	
3		9	1	8	0	0	

#	Rob	P	KV	\$F	PM	I	Bemerkungen
22.11.							
1	K1 r	9	1	8	0	0	mit eingebauter Totzeit, instabil
2		9	1	8	99	0	mit eingebauter Totzeit, instabil
30.11.							
1	K1 r	1	1	0	0	0	
2		1	0	0	0	0	
3		5	1	0	0	0	
4		5	0	0	0	0	
5		9	1	0	0	0	
6		9	0	0	0	0	
7.12.							
1	K1 r	11	1	0	0	0	
2		12	1	0	0	0	neues Gütefunktional
3		13	1	0	0	0	
4		--	1	0	--	0	
5		--	0	0	--	0	
8.12.							
1	K1 r	14	0	8	0	0	MW-Filter integriert
2		15	0	8	0	0	
3		16	0	8	0	0	
4		14	0	8	99	0	instabil
5		15	0	8	99	0	dito
6		15	0	8	99	0	dito
9.12.							
1	K2 r	--	0	4	--	0	Chirp x, (am IWB)
2		--	0	4	--	0	Chirp y
3		--	0	4	--	0	Chirp z
4		--	0	4	--	0	Chirp α

#	Rob	P	KV	\$F	PM	I	Bemerkungen
5	K2 r	--	0	4	----	0	Chirp β
6		--	0	4	--	0	Chirp γ
7		--	0	4	--	0	Chirp α , $A=1\text{mrad}$, $k=0,5\text{s}^{-2}$, $f=7\text{ Hz}$
8		--	0	4	--	0	Chirp β , <i>dito</i>
9		--	0	4	--	0	Leere Bahn, β von Hand anregen
10		17	0	4	0	0	Abbruch
10.12.							
1	K1 r	18	0	4	0	0	
2		18	0	0	0	0	
3		18	0	4	99	0	Abbruch
4		20	0	4	0	0	
5		--	0	0	0	0	Chirp γ
6		--	0	4	0	0	Chirp γ
7		11	0	0	0	0	
8		21	0	0	0	?	
11.12.							
1	K1 r	21	0	0	0	0	
2		--	0	0	--	0	später/auch 0 genannt
3		22	0	0	0	0	
14.12.							
1	K1 r	23	0	0	0	0	
2		23	0	0	99	0	Abbruch
3		24	0	0	0	0	
4		25	0	0	0	0	
5		26	0	0	0	0	
6		28	0	0	0	0	
7		29	0	0	0	0	

Tabelle 14: Versuchsprotokoll

Hierbei sind:

- Rob** Roboterkonfiguration (r=real, s=simuliert)
- P** Parametersatz (s. Tabelle 13)
- KV** Kompensationsvorsteuerung aktiviert (1) oder deaktiviert (0)
- \$F** Mittelwertfilterbreite (0=deaktiviert)
- PM** Prädiktionsmodus (0=Prädiktion, 99=Rückkopplung, s. §3.2.3)
- I** Integralregler (1=aktiviert, 0=ab Schritt 0 deaktiviert)

Die Experimente sind in Ordnern nach Datum (Format *JJJJMMTT*) und Versuchsnummer geordnet im Verzeichnis „*Experimente*“ zu finden. Die *.pose-Dateien haben folgendes Format:

Spalte(n)	Formelzeichen	Bezeichnung / Bemerkung
1		Zeitvektor
2 – 7	${}_R \underline{p}_{0,scp}^{soll} \hat{=} w$	Solllage des SCP („SOLL“)
8 – 13	${}_R \underline{p}_{0,scp0}^{ist} \hat{=} u^{ist}$	Istlage des SCP0 („MESS“)
14 – 19	${}_R \underline{p}_{scp0,scp}$	Auslenkung SCP rel. zum SCP0
20 – 25	${}_R \underline{p}_{tcp0,tcp}$	Auslenkung TCP rel. zum TCP0
26 – 31		Kommandierter Befehl („KOMM“)
32 – 37	${}_R \underline{p}_{0,scp}^{soll'} \hat{=} w'$	modifizierte Solllage des SCP
38 – 43		Stellsignal der präd. Steuerung ($=u^{soll} \leftrightarrow$ kein I-Regler)
44 – 49	$\tilde{y}^{präd}$	prädizierter Systemausgang
50 – 55	${}_R \underline{p}_{0,scp0}^{soll} \hat{=} u^{soll}$	Solllage des SCP0 („SOLL ARM“)

Tabelle 15: Format der pose-Dateien

Die etwas chaotisch anmutende Anordnung ist über die Zeit gewachsen und wurde aus Gründen der Abwärtskompatibilität beibehalten. Die Dateien können über die `load`-Befehl ohne Modifikation in MATLAB® eingelesen werden.

Anhang F: Sonstiges

Verzeichnisstruktur der CD

Experimente

 **Datensätze** → *siehe Tabelle 13*


 **Parametersätze** → *siehe Tabelle 14*

Programme

Matlab

 Auswertung *Vergleich von verschiedenen Experimenten*

 Identifikation *Systemidentifikation (→ Kapitel 4.1)*

 Simulation *Simulation der Steuerung in MATLAB*


 Sonstiges *u.a. alte Simulationen*

 **Rahmen** *Gesamtversion von `rahmen` mit Steuerung*

 rsmk2pose *Konvertierung von `rsmk.dat` zu `*.pose`*

 Chirp *Erstellung von Chirp-Dateien*

Sonstiges

 **Ausarbeitung** *diese Ausarbeitung*

 **Literatur** *Sammlung von Literatur mit Übersicht*

Quellenverzeichnis

- [1] O. Föllinger: „Regelungstechnik: Einführung in die Methoden und ihre Anwendung“, Hüthig, 1994
- [2] J. Lunze: „Regelungstechnik 1“, 2010, 8. Auflage, Springer
- [3] R. Dorf, R. Bishop: "Moderne Regelungssysteme", 10. Auflage, Pearson 2006
- [4] B. A. Shenoi: „Introduction to Digital Signal Processing and Filter Design“, Wiley-Interscience, 2006, S. 250
- [5] B. Widrow, J. M. McCool, B. P. Medoff: „Adaptive control by inverse modeling“, Proceedings of the 12th IEEE Asilomar Conference on Circuits, Systems and Computers, S. 90-94, 1978
- [6] L. Silverman: „Inversion of multivariable linear systems“, IEEE Transactions on Automatic Control, vol. AC-14, no. 3, S. 270-276, 1969
- [7] S. Devasia, D. Chen, B. Paden: „Nonlinear inversion-based output tracking“, IEEE Transactions on Automatic Control, Vol. 41, No.7, S. 930-42, 1996
- [8] Q. Zou, S. Devasia: „Preview-based stable-inversion for output tracking“, Proceedings of the American Control Conference 1999, Vol. 5. S. 3544-3548, 1999
- [9] S. Devasia: „Should Model-Based Inverse Inputs Be Used as Feedforward Under Plant Uncertainty?“, IEEE Transactions on Automatic Control, Vol. 47, No. 11, 2002, S. 1865ff
- [10] J. T. Wen, B. Potosaid: „An Experimental Study of a High Performance Motion Control System“, American Control Conference, Boston 2004, S. 5158-5163
- [11] C.-H. Menq, J.J. Chen: „Precision Tracking Control of Discrete Time Nonminimum-Phase Systems“, American Control Conference, 1992, S. 1097ff
- [12] M. Ruzbehani: „A new Tracking Controller for discrete-time SISO non-minimum phase Systems“, Asian Journal of Control, Vol. 12, No. 1, S. 89-95, 2010
- [13] A. Kamel: „Dämpfung von Schwingungen eines Industrieroboters mit nachgiebigem Kraft-Momentensensor“, Bachelorarbeit, Lehrstuhl für Steuerungs- und Regelungstechnik, TU München, Prof. Buss, 2007
- [14] E. F. Camacho, C. Bordons: „Model Predictive control“, Springer, 2004
- [15] R. Dittmar, B.-M. Pfeiffer: „Modellbasierte prädiktive Regelung“, Oldenbourg Wissenschaftsverlag GmbH, 2004

-
- [16] S. J. Qin, T. A. Badgwell: „A survey of industrial model predictive control technology“, Control Engineering Practice, Vol. 11, S. 733–764, 2003
 - [17] M. Husty, A. Krager, H. Sachs, W. Steinhilper: „Kinematik und Robotik“, Springer, 1997, S. 417ff
 - [18] S. B. Niku: „Introduction to Robotics: Analysis, Control, Applications“, John Wiley & Sons Inc., 2011, S. 44ff
 - [19] F. Lange, „Montage in Bewegung“, DLR-interne Dokumentation, Sept. 2010
 - [20] J. Maciejowski: „Predictive Control with Constraints“, Prentice Hall 2002, S. 77
 - [21] R. Dittmar, B.-M. Pfeiffer: „Modellbasierte prädiktive Regelung“, Oldenbourg Wissenschaftsverlag GmbH, 2004, S. 139
 - [22] J. Maciejowski: „Predictive Control with Constraints“, Prentice Hall 2002, Kap. 6
 - [23] J. Maciejowski: „Predictive Control with Constraints“, Prentice Hall 2002, S. 191ff
 - [24] D. W. Clarke, C. Mohtadi, P.S. Tuffs: “Generalized Predictive Control Part I. - The Basic Algorithm“, Automatica, Vol. 23, No. 2, pp. 137-148, 1987
 - [25] J. M. Maciejowski: „Predictive Control with Constraints“, Prentice Hall 2002, S. 6
 - [26] D. Meschede: „Gerthsen Physik“, Springer, 22. Auflage, S. 181ff
 - [27] Lennart Ljung: „System Identification - Theory For the User“, 2. Ausgabe, PTR Prentice Hall, 1999
 - [28] T. Thümmel, „Vorlesung Experimentelle Schwingungsanalyse“, Skriptum der TU München, Lehrstuhl Angewandte Mechanik (Prof. Ulbrich), Stand 2/07
 - [29] <http://www.mathworks.com/help/techdoc/index.html>
 - [30] W. Schiehlen, P. Eberhard: „Technische Dynamik: Modelle für Regelung und Simulation“, 2. Auflage, Teubner, Sept. 2004, S. 28ff